USER MANUAL

CM4000 and CMx86 Family

User Manual

Rev: 1.6

November 17, 2005

CHAPTER

1. Introduction
2. Installation
3. System Configuration
4. Configuring Serial Ports
5. Configuring Dial-In Access
6. Secure Remote Desktop Connection
7. Logging and Alerts
8. Power Control
9. Authentication
10. System Management
11. Basic Configuration - Linux Commands
12. Advanced Configuration

APPENDIX

A. Linux Commands
B. Hardware Specification
C. Safety and Certifications
D. Connectivity and Serial I/O
E. Hardware Test
F. Terminology
G. End User License Agreement
H. Service and Warranty

Chapter 1 Introduction

This Manual

This Users Manual walks you through installing and configuring your CM4008, CM4116, CM4148 or CMx8604 console server (referred to generically in the manual as CM4000) and. Once configured, you will be able to use your CM4000 to securely control your network routers, as well as the servers and power strips in your data center; and securely connect to PC systems in smaller remote offices. This manual guides you in managing this infrastructure locally (across your LAN or through the local console port), and remotely (across the Internet or *via* dial up).

Chapter 2 of this manual covers the physical installation of your CM4000 console server and the interconnecting of controlled devices. It also covers the initial software installation required to build a CMx86 console server.

Each CM4000 hosts a web server with a web based Management Console that allows you the Administrator to configure the console server with your browser. Chapters 3 to 10 of this manual take you through installation and configuration using this Management Console. It also instructs you on setting up User accounts for those other external users to whom you want to give serial port access.

The CM4000 runs an embedded Linux operating system. Experienced Linux and UNIX users may prefer to undertake configuration at the command line. You can get command line access by connecting through a terminal emulator or communications program to the console serial port, or by *telnet* connecting through the LAN. Chapter 11 takes you through installation and configuration using the *config* command.

The Management Console and the *config* command enable you to complete the configurations you typically will require. Chapter 12 covers more advanced configuration activities, where you will need to use Linux commands to edit config files *etc*.

The latest update of this manual can be found online at
www.opengear.com/download.html

Manual Conventions

This manual uses different fonts and typefaces to show specific actions:

Note Text presented like this indicates issues you should take note of.

WARNING
Text presented like this highlights important issues.
It is essential you read and take head of these warnings.

- Text presented with an arrow head indent indicates an action you should take as part of the procedure.

Bold text indicates text that you type, or the name of a screen object (*e.g.* a menu or button) on the Management Console.

Italic text indicates a text command to be entered at the Linux kernel command line level.

Chapter 2 Installation

Introduction

This chapter describes the physical installation of the CM4000 console server hardware and interconnection to the network and controlled appliances.

WARNING
To avoid physical and electrical hazard
please read Appendix C on Safety

Models

There are three models of the CM4000, each with a different number of serial ports:

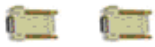
- CM4008 – eight ports
- CM4116 – sixteen ports
- CM4148 – forty eight ports
- CMx86 – four ports (upgradeable to sixty four ports)

The tables below show the component shipped with each model. Check you have all the parts listed before you begin the installation and configuration.

CM4008 Kit Components

Part # 509000 CM4008 Console Server

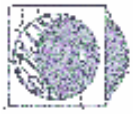
Part # 440016 2 x Cable UTP Cat5 blue



Part # 319000 and 319001 Connector DB9F-RJ45S straight and DB9F-RJ45S cross-over

Part # 450006 Power Supply 5VDC 2.0A IEC Socket

Part # 440001 IEC AC power cable



Part #539000 Quick Start Guide and CD-ROM

- Unpack your CM4008 Kit and verify you have all the parts shown above, and that they all appear in good working order
- Proceed to connect your CM4008 to the network, the serial ports of the controlled servers and AC power as shown below

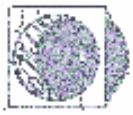
CM4116 (or CM4148) Kit Components

Part # 509001 (or Part # 509002) CM4116/4148 Console Server

Part # 440016 2 x Cable UTP Cat5 blue

Part # 319000 and 319001 Connector DB9F-RJ45S straight and DB9F-RJ45S cross-over

Part # 440001 IEC AC power cord



Part # 539001 Quick Start Guide and CD-ROM

- Unpack your CM4116 (or CM4148) Kit and verify you have all the parts shown above, and that they all appear in good working order

- If you are installing your CM4116 or CM4148 in a rack you will need to attach the rack mounting brackets supplied with the unit, and install the unit in the rack. Take care to head the Safety Precautions listed in Appendix C
- Proceed to connect your CM4008 to the network, the serial ports of the controlled servers, and AC power as detailed below

Power connection

The CM4116 and CM4148 models have a universal auto-switching AC power supply built in. This power supply accepts AC input voltage between 100 and 240 VAC with a frequency of 50 or 60 Hz and the power consumption is less than 20W.

Both CM4116 and CM4148 models have an IEC AC power socket located at the rear of the metal case. This IEC power inlet uses a conventional IEC AC power cord, and the power cord for North American is provided by default. There is a warning notice printed on the back of each unit:

WARNING
To avoid electrical shock the power cord grounding conductor must be connected to ground.

The CM4008 is supplied with an external DC power supply unit. This unit accepts an AC input voltage between 100 and 250 VAC with a frequency of 50 or 60 Hz. The DC power supply has an IEC AC power socket, which accepts a conventional IEC AC power cord. The power cord for North American is provided by default. The 5V DC connector from the power supply plugs into the 5VDC power socket on the rear of the CM4008 chassis.

- Plug in the AC power cable (and the DC power cable for CM4008) and turn AC power On
- Confirm the Power LED on the front of the panel is lit. (Note: When you have applied power to the CM4008, you will also observe the LEDs P1 through P8 light up in sequence)

Network connection

The RJ45 LAN port is located on the rear panel of the CM4008, and on the front panel of the rack-mount CM4116 and CM4148. All physical connections are made using industry standard Cat5 cabling and connectors. Ensure you only connect the LAN port to an Ethernet network that supports 10Base-T/100Base-T.

Serial Port connection

The RJ45 serial ports are located on the rear panel of the CM4008 and on the front panel of the rack mount CM4116 and CM4148. Likewise the DB9 LOCAL (Console/Modem) port is on the rear of the CM4008 and the front of the CM4116 and CM4148.

Conventional Cat5 cabling with RJ45 jacks are used for all serial connections. Before connecting the console port of an external device to the CM4000 serial port, confirm that the device does support the standard RS-232C (EIA-232).

Opengear supplies a range of cables and adapters that may be required to connect to the more popular servers and network appliances. These are overviewed in Appendix D (Connectivity and Serial I/O). More detailed information is available online at http://www.opengear.com/cabling.html

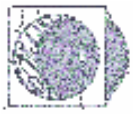
Note Care should be taken in handling CM4000 products. There are no operator serviceable components inside, so please do not remove covers, and do refer service to qualified personnel.

CMx8604 Kit Components

MP4056A 4 port multiport card

DB37M to 4 Ports DB9M

Multiport board driver CD & Sunix User Manual



CMx8604 Quick Start Manual & CD-ROM,

- Unpack your CMx8604 kit and verify you have all the parts shown above, and that they all appear in good working order
- The CMx86 console server is built by installing CMx86 software and the MP4056A multiport card in a standard x86 PC system. This x86 system can have any operating system running, and CMx86 should work with most network cards. The system must have an IDE hard disk (or at least a USB flash drive in order to store it's configuration between boots).
- Insert the 4 port multiport card into a free PCI bus slot in your x86 system. For CMx86 operation you do not have to install any of the Sunix software
- Insert the CMx86 CD and apply power to the x86 system (to boot from the CD)
- On initial boot up, you will be presented with a menu asking you to select if your VGA screen or your COM port 1 is to be used as the “local console port” for your CMx86 system. By default the VGA screen will be used. The console menu will also appear on COM1 at 9600 baud for headless operation.
- The Linux kernel will then load and the CMx86 application will automatically recognize, and appropriately number, all the installed serial ports in your system (*i.e.* the MP4056 ports and any other standard COM ports installed)
- You must log in, at the Linux command line level, using the user name ***root*** and the password ***default.*** You will then be presented with a command line prompt (which is a hash #)

Note You can access the Linux command line using the "local console port" you selected when you booted from the CD (the VGA console or COM1 @ 9600). You can also telnet to the device at IP address 192.168.0.1.

- If you wish to install the CMx86 on the hard disk so that the x86 system boots as a dedicated console server, type ***setup-hd <device>***

  This *setup-hd* command will install the CMx86 software onto the nominated IDE (or flash) hard drive <*device*> in your system. It will erase any information on that drive and completely takes over the drive.

  If you run the *setup-hd* command with no arguments it will list all the hard drives that it has found.

  The *setup-hd* command will also ask you if you want to create a new config file system. You should answer '***y***' (the default) here the first time through. (You can also use the *setup-hd* command to upgrade - as updated releases become available- by answering 'n' to the question about creating a new config file system.)

  If, when you booted from the CD you selected a serial local console, then the hard disk will boot with a serial console. If you selected a VGA console and install to hard disk, the hard disk will boot with a VGA console

- Alternately if you want to run the CMx86 console server from the CD, and use a USB/flash device to store all the config file information then type ***setup-flashkey <device>***

  This *setup-flashkey* command will format the flash key for use as a config file system and completely erases any information on the flash key.

  If you run *setup-flashkey* with no arguments it will list any USB/flash drives that it has found.

  USB config file systems have priority over all others. If you install to hard disk then you can override the config file system on the HD by connecting a formatted USB key to the system.

- Proceed now to Chapter 11, if you wish to continue configuring your CMx86 console server and setting up ports and users from the Linux command line, or to Chapter 3 for configuration using a browser

Chapter 3 System Configuration

Introduction

This chapter provides step-by-step instructions for installing your Opengear CM4000 console server into your network and connecting to the Internet. This involves:

- Activating the Management Console
- Changing the administration password
- Setting the IP address and
- Selecting network services

Connect to the Management Console

Your CM4000 comes configured with a default IP Address 192.168.0.1 Subnet Mask 255.255.255.0

- Directly connect a PC or workstation to the CM4000

Note For simplicity, during initial connection it is recommended that the CM4000 console server is connected directly to a single PC or workstation. However, if you choose to connect your LAN before completing the initial setup steps, it is critical that:

- you ensure there are no other devices on the LAN with an **address of 192.168.0.1**
- the console server and the PC/workstation are on the same LAN segment, with no interposed routers or gateway appliances

- To browser configure the CM4000; the connected PC or workstation should have an IP address in the same range as the CM4000. If this is not convenient, you

can use the ARP-Ping command as described in the Note below to reset the CM4000 IP address

Note The PC/workstation must have an address in the same network range as the CM4000 (e.g. 192.168.0.100).

To configure the IP Address of your Linux or Unix PC/workstation simply run *ifconfig*. For Windows PCs (Win9x/Me/2000/XP/ NT):

- Click **Start** -> (**Settings** ->) **Control Panel** and double click **Network Connections** (for 95/98/Me, double click **Network**).
- Right click on **Local Area Connection** and select **Properties**
- Select **Internet Protocol (TCP/IP)** and click **Properties**
- Select **Use the following IP address** and enter the following details:
  IP address: **192.168.0.100**
  Subnet mask: **255.255.255.0**

  If you wish to retain your existing IP settings for this network connection, click **Advanced** and **Add** the above as a secondary IP connection.

- Activate your preferred browser on the connected PC/ workstation and enter **http://192.168.0.1** The Management Console can be used with all current versions of the popular browsers (Netscape 7.0 and 6.0, Internet Explorer 6.0, Mozilla Firefox 1.0 and more)

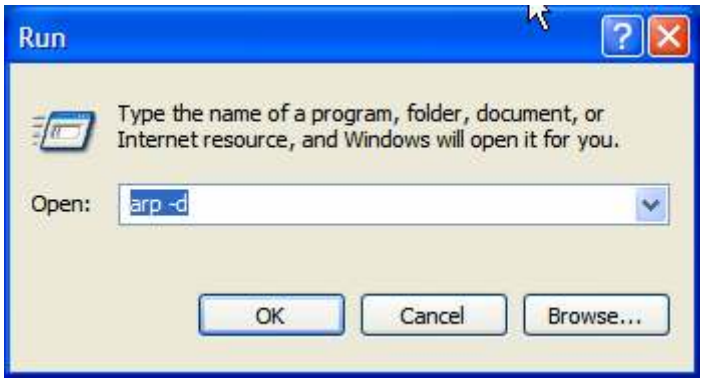
- You will be prompted to log in. Enter the default administration user name and administration password:

  Username: **root**
  Password: **default**

Note ARP-Ping IP Address Assignment

An alternative connection option is to use the *arp* command on a network connected PC/workstation to assign an alternate starting IP address to the CM4000. To do this from a Windows PC:

- Click **S**tart -> R**un**
- Type *cmd* and click **OK** to bring up the command line
- Type *arp –d* to flush the ARP cache
- Type *arp –a* to view the current ARP cache which should be empty



Now add a static entry to the ARP table and ping the CM4000 to have it take up the IP address. In the example below we have a CM4000 unit with a MAC Address 00:13:C6:00:02:0F (designated on the label on the bottom of the unit) and we are setting its IP address to 192.168.100.23:

- Type *arp -s 192.168.100.23 00-13-C6-00-02-0F*
- Type *ping -t 192.18.100.23* to start a continuous ping to the new IP Address.
- Turn on the CM4000 and wait for it to configure itself with the new IP address. The Opengear will start replying to the ping at this point

OpenGear Management Console: Welcome
Network
IP
Dial-In
Services
Serial Port
Configuration
Users
Trusted Networks
Alerts & Logging
Event Log
Alerts
Syslog
Welcome to the OpenGear Management Console!
You will need to configure the following in order to have a usable Opengear unit. After completing a step by following the apropriate link you can return to the updated configuration steps by clicking on the logo in the top left corner of the Management Console.
1. Change the default administration password on the Administration/System page..
2. Configure the local network settings on the Network/IP page..
3. Configure serial ports settings and enable supported protocols on the Serial Port/Configuration page. Done.
4. Configure users with access to serial ports on the Serial Port/Users page. Done.

A **Welcome** screen, which lists the four basic installation configuration steps, will be displayed:

1. **Change the default administration password ...** (Covered in this Chapter 3)
2. **Configure the local network settings ...** (Covered in this Chapter 3)
3. **Configure serial ports settings ...** (Refer Chapter 4 - Configuring Serial Ports)
4. **Configure users with access ...** (Refer Chapter 4 - Configuring Serial Ports)

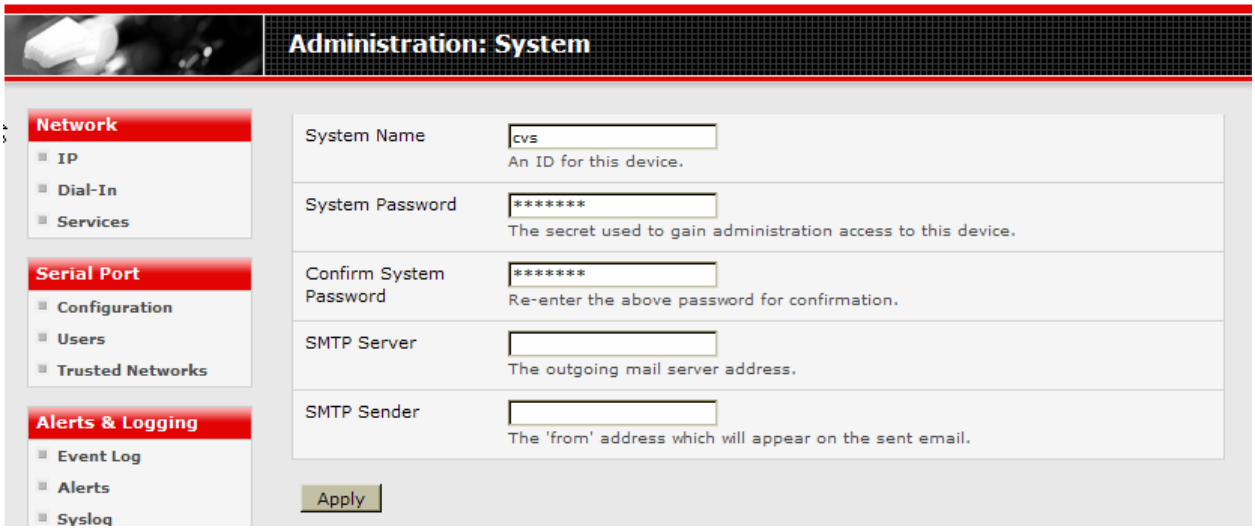
After completing each of the above steps, you can return to the configuration list by clicking on the **Opengear** logo in the top left corner of the screen. As you complete each step, the configuration list will be updated *e.g.* after you have configured the serial ports it will display this step as:

3. **Configure serial ports settings and enable remote access method on the *Serial Port/Configuration* page. Done.**

Note If you are not able to connect to the Management Console at 192.168.0.1 or if the default Username / Password were not accepted then reset your CM4000 as described in Chapter 9.

Change the default Password

For security reasons, only the administration user (the "Administrator") named **root** can log into your console server. So only those people who know the root password can access and reconfigure the server. The corollary is that anyone who correctly guesses the root password could gain access (and the default root password is **default**). Given this, it is essential you enter and confirm a new password for your CM4000 before giving it any access to control of your appliances.

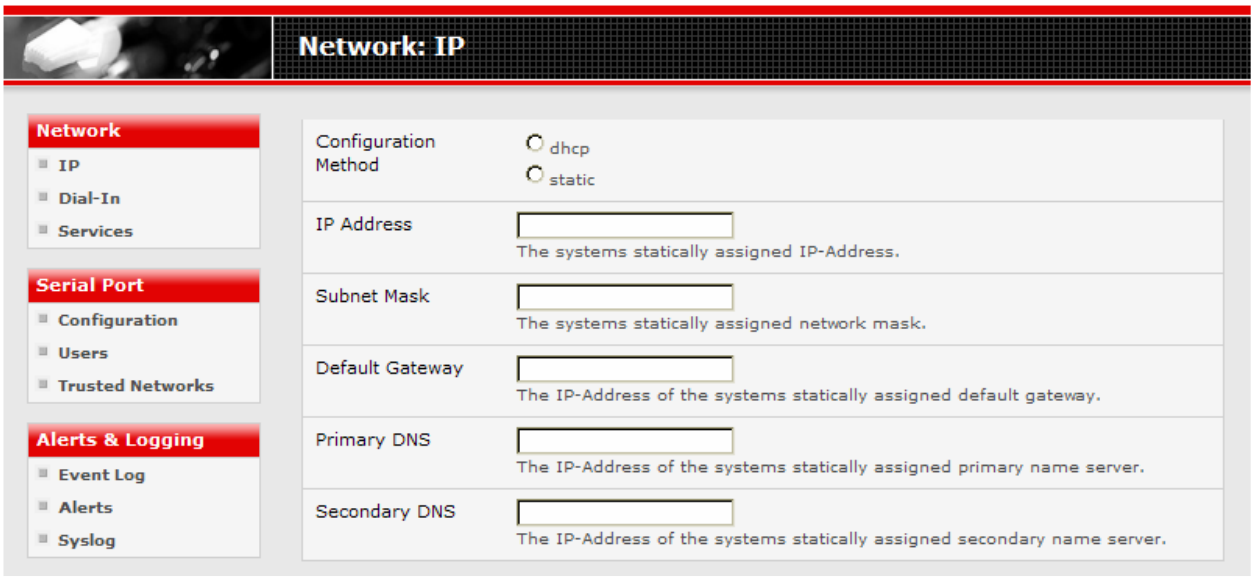


- Select the **Administration: System** menu option
- Enter a new **System Password** then re-enter it in **Confirm System Password.** This is the new password for **root**, the main administrative user account, so it is important that you choose a password that is hard to guess, and keep it safe
- At this stage you may also wish to enter **System Name** to give the console server a unique ID
- Click **Apply**. As you have changed the password you will be prompted to log in again. This time use the new password

Note If you are not confident your CM4000 has been supplied with the current release of firmware, you can upgrade. Refer Chapter 9.

Setting the console server's Network IP address

You now must specify if the console server is to be manually assigned an IP address, or if it is to automatically obtain an address from a DHCP server on the network.



- Select the **Network: IP** option
- Select **dhcp** or **static** for the **Configuration Method**
- If you selected **static** you must manually enter the new IP Address, Subnet Mask, Default Gateway and DNS Server. This selection automatically turns off the DHCP client

- If you selected **dhcp,** the CM4000 will look for configuration details from a DHCP server on your LAN. This selection automatically disables any static address. The CM4000 MAC address can be found on a label on the base plate.

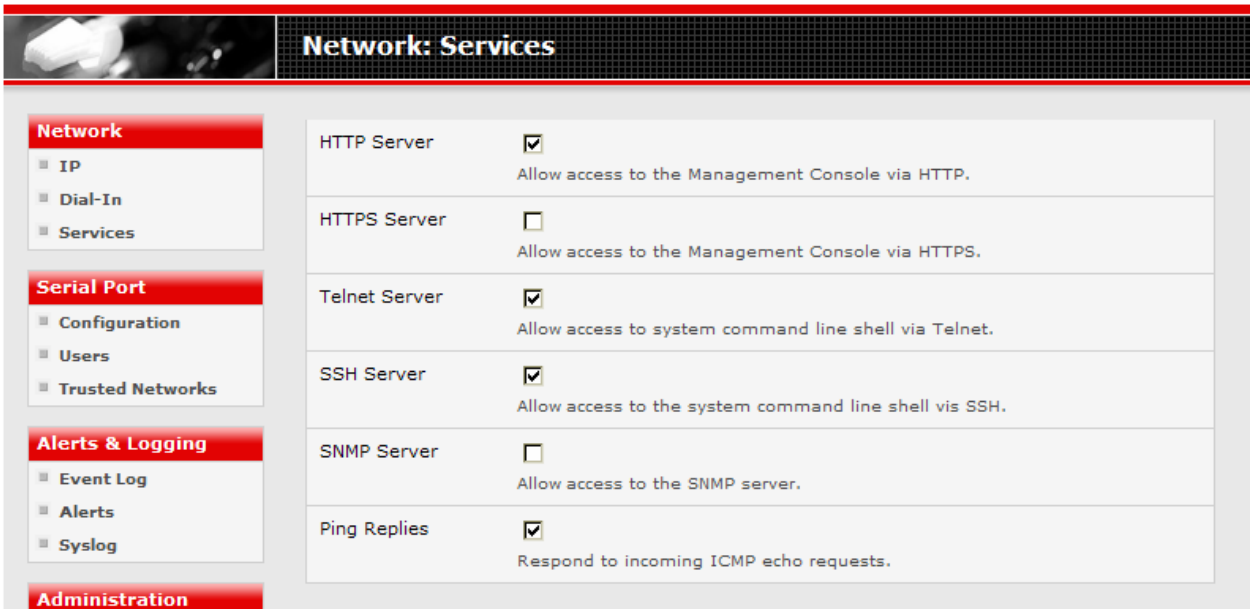
> **Note** In its factory default state (with no Configuration Method selected) the CM4000 has its DHCP client enabled, so it will automatically accept any network IP address assigned by the DHCP server on your network. The CM4000 will then respond to both its Static address (192.168.0.1) and its DHCP address.

- Click **Apply**
- Reconnect the browser on the PC/workstation that is connected to the CM4000 by entering **http://new IP address**

> **Note** If you have changed the CM4000 IP address, you may need to reconfigure your PC/workstation so it has an IP address that is in the same network range as this new address (as detailed in an earlier note in this chapter).

Select appropriate Network Services

The CM4000 has a broad range of network access and related services that need to be enabled or disabled. The factory default configuration enables HTTP, Telnet, SSH and Ping and disables HTTPS and SNMP. You can modify this very simply to disable any of the services, or enable others:



- Select the **Network: Services** option then select (or deselect) the tick-box for the service to be enabled (or disabled)
- Click **Apply**. As you apply your services selections, the screen will be updated with a confirmation message:

  Message Changes to configuration succeeded.

The following service options are available:

HTTP This allows the Administrator (user named **root**) basic browser HTTP access to the Management Console. By default both HTTP and HTTPS are enabled, however either or both can be disabled. It is recommended this service be disabled if the CM4000 is to be remotely administered over the Internet.

HTTPS This allows secure HTTP access to the Management Console. If you enable HTTPS, the Administrator will be able to use a secure browser connection to the CM4000 Management Console. For information on certificate and user client software configuration refer Chapter 9 - Authentication. By default HTTPS is enabled, and it is recommended that only HTTPS access be enabled if the console server is to be managed over the Internet.

Telnet This gives the administration user Telnet access to the system command line shell (Linux commands). While this is suitable for local direct connection, it is recommended this service be disabled if the CM4000 is to be remotely administered.

SSH This provides secure SSH access to the Linux command line shell. It is recommended you choose SSH as the protocol where the administrator is connecting to the console server over the Internet or other public network. This will provide authenticated SSH communications between the SSH client program on the remote administrator's PC/workstation and the console server. For more information on SSH configuration refer Chapter 9 - Authentication.

SNMP This will enable *netsnmp* in the console server, which will keep a remote log of all posted information. To modify the SNMP settings, the Administrator must make the edits at the command line as described in the Chapter 10 - Advanced.

Ping This allows the CM4000 to respond to incoming ICMP echo requests. For security reasons this service is generally disabled.

Chapter 4 Configuring Serial Ports

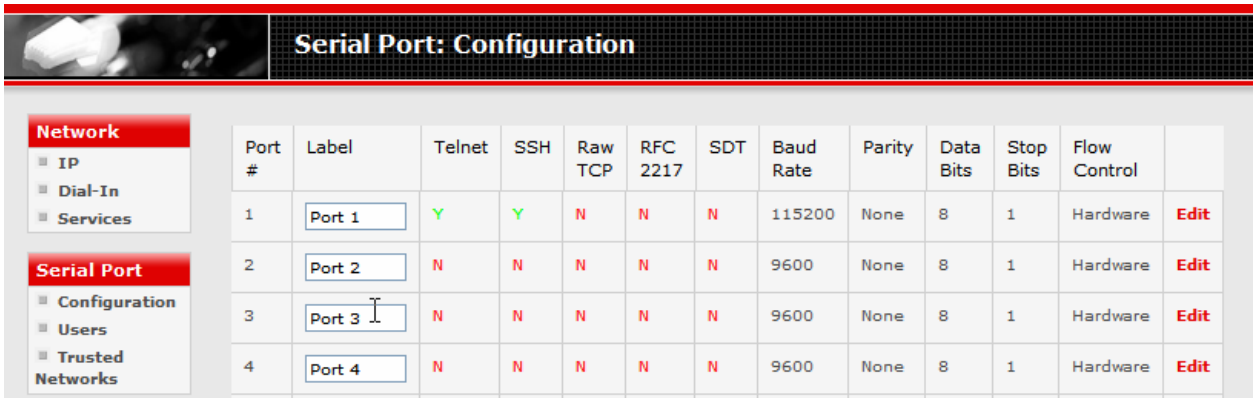
Introduction

Before configuring the serial ports, you should connect the ports to the serial devices they will be controlling. Then configure the port, which entails:

- Giving each port a “label” (optional)
- Configuring the serial communications protocols to be used by each port
- Setting the serial RS232 parameters for each serial port
- Setting up new users (with nominated access rights to these ports)
- Specifying where those users have to be located to have access to the ports

Port Labels

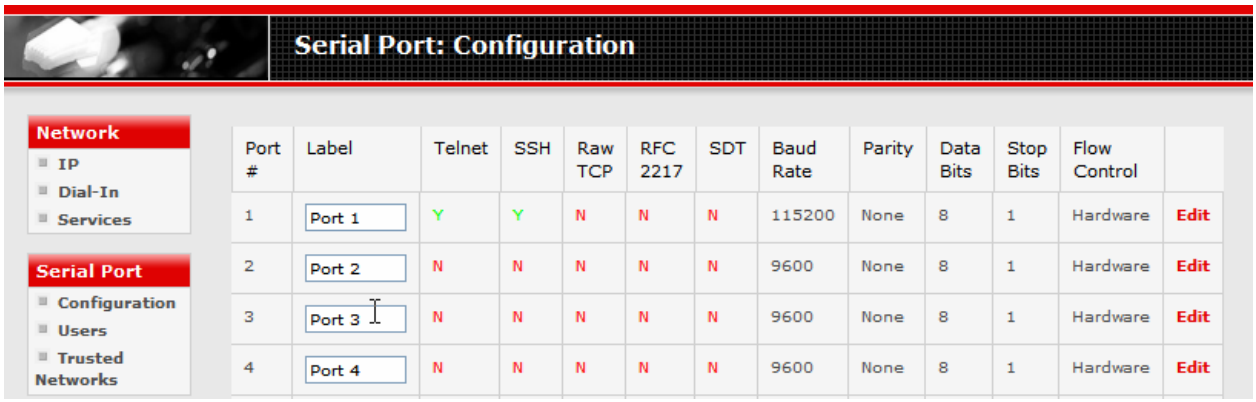
The first step is to assign a user label on each port, for easy management:


Serial Port: Configuration
Network
IP
Dial-In
Services
Serial Port
Configuration
Users
Trusted Networks
Port # | Label | Telnet | SSH | Raw TCP | RFC 2217 | SDT | Baud Rate | Parity | Data Bits | Stop Bits | Flow Control
1 | Port 1 | Y | Y | N | N | N | 115200 | None | 8 | 1 | Hardware | Edit
2 | Port 2 | N | N | N | N | N | 9600 | None | 8 | 1 | Hardware | Edit
3 | Port 3 | N | N | N | N | N | 9600 | None | 8 | 1 | Hardware | Edit
4 | Port 4 | N | N | N | N | N | 9600 | None | 8 | 1 | Hardware | Edit

➢ Select the **Serial Port: Configuration** menu option, and enter the desired **Label** for each port

Protocol Configuration

The next step is to set up the communications protocol. For each serial port you have a selection of protocol options that can be used when you "connect" to that port:



Port #	Label	Telnet	SSH	Raw TCP	RFC 2217	SDT	Baud Rate	Parity	Data Bits	Stop Bits	Flow Control	
1	Port 1	Y	Y	N	N	N	115200	None	8	1	Hardware	Edit
2	Port 2	N	N	N	N	N	9600	None	8	1	Hardware	Edit
3	Port 3	N	N	N	N	N	9600	None	8	1	Hardware	Edit
4	Port 4	N	N	N	N	N	9600	None	8	1	Hardware	Edit

- Select the **Serial Port: Configuration** menu option
- To change the Port Configuration click **Edit**

> **Note** The factory default setting has Telnet, SSH and RAW TCP all disabled. So you must select **Telnet**, **SSH**, **RAW TCP**, **RFC2217** or **SDT** for each port you will be using.

Telnet With Telnet, the Telnet client on the user's PC/workstation connects to the serial device attached to the console server. The communications are unencrypted so this protocol is recommended for local connections

For a Telnet connection from Win2000/XP/ NT, you can run *telnet* from the command prompt (*cmd.exe*). *PuTTY* also supports Telnet. The port address is IP Address _ Port (2000 + serial Port #) *i.e.* 2001 – 2048

SSH It is recommended that you choose SSH as the protocol where the user is connecting to the console server over the Internet or other public network. This will provide authenticated SSH communications between the SSH client program on the remote user's PC/workstation and the console server, so the user's communication with the serial device attached to the console server is secure

For SSH, use *PuTTY* or *SSHterm,* and the port address is IP Address _ Port (3000 + serial Port #) *i.e.* 3001 – 3048

TCP RAW TCP allows connections directly to a TCP socket. However while communications programs like *PuTTY* also supports RAW TCP, this protocol would usually be used by a custom application

For RAW TCP, the port address is IP Address _ Port (4000 + serial Port #) *i.e.* 4001 – 4048

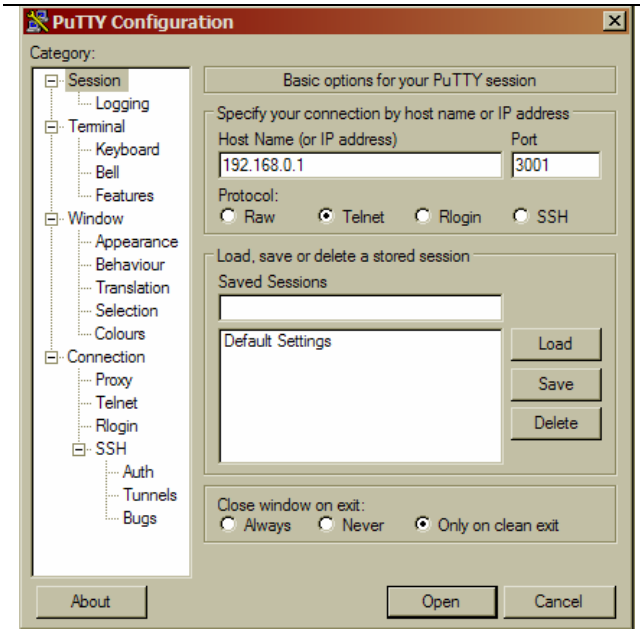
RFC2217 Selecting RFC2217 enables serial port redirection on that port. Special client software is available for Windows UNIX and Linux that supports RFC2217 virtual com ports (see the *CM4000 Serial Port Redirector Client* section below for details)

SDT Secure Desktop Tunneling allows secure tunneling of Microsoft's Remote Desktop Protocol (RDP) connections. For SDT Remote Desktop (RDP), the port address is IP Address _ Port (7300 + serial Port #) *i.e.* 7301 – 7348. For SDT VNC, the port address is IP Address _ Port (7900 + serial Port #) *i.e.* 7901 – 7948. And for SSH SDT the port address is IP Address _ 22. Refer to the *Secure Desktop Tunneling-Serial* chapter for details

Note Depending on the protocols selected, you may also need to configure appropriate communications software on each user's PC/workstation.

PuTTY

One useful communications package is *PuTTY,* a freeware implementation of Telnet and SSH for Win32 and UNIX platforms. It runs as an executable application without needing to be installed onto your system. *PuTTY* (the Telnet and SSH client itself) can be downloaded at http://www.tucows.com/preview/195286.html



- To use PuTTY for an SSH terminal session from a Windows Client, you type a 'Host Name (or IP address)' and 'TCP port' number. In the case of the CM4000, the TCP port will be 3000 plus the physical serial port number (*i.e.* 3001 to 3048). Then select 'SSH' as the protocol. Click the 'Open' button. You may then receive a 'Security Alert' that the host's key is not cached, you will need to choose 'yes' to continue. You will then be presented with the login prompt of the remote system connected to the serial port chosen on the CM4000 device. You can login as normal and use the host serial console screen.

- The procedure to set up a Telnet session is similarly simple and the PuTTY Configuration screen (above) shows the connection set up to Telnet to Port 1 (2001) on a console server with an IP address of 192.168.0.1

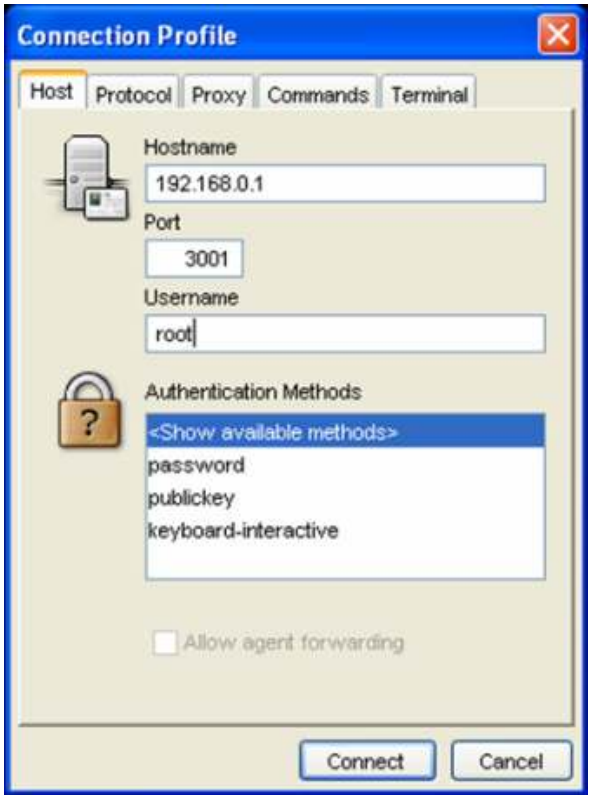
SSHTerm

Another useful communications package is *SSHTerm,* an open source package that can be downloaded from http://sourceforge.net/projects/sshtools

- To use SSHTerm for an SSH terminal session from a Windows Client you simply Select the 'File' option and click on 'New Connection'.

- A new dialog box will appear for your 'Connection Profile' where you can type in the host name or IP address (for the CM4000 unit)and the TCP port that the SSH session will use. In the case of the CM4000, the TCP port will be 3000 plus the physical serial port number (*i.e.* 3001 to 3048). Then type in your user name and choose password authentication and click connect.

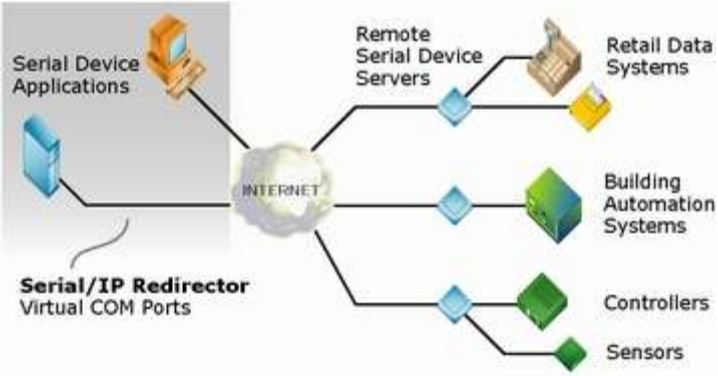
You may receive a message about the host key fingerprint, and you will need to select 'yes' or 'always' to continue. The next step is password authentication and you will be prompted for your user name and password from the remote system. You will then be logged into the remote system connected to the serial port chosen on the CM4000 device and presented with its serial console screen.



CM4000 Serial Port Redirector Client

To access the virtual serial ports that RFC2217 support, you need to run client software (to actually redirect local serial ports to remote CM4000 serial ports).

For Windows, Opengear recommends the Serial/IP™ COM Port Redirector from Tactical Software, which creates virtual COM ports for applications to use serial device servers without software changes. Tactical Software provides a trial copy of its products



For Linux, AIX, HPUX, SCO, Solaris and Unixware, Opengear has released an open source ***opengear-serial-client*** utility, which can be freely downloaded. This serial port redirector software is loaded in your desktop PC, and it allows you to use a serial device connected to the remote CM4000 as if it were connected to your local serial port. *opengear-serial-client* creates a pseudo tty port, connects the serial application to the pseudo tty port, receives data from the pseudo tty port, transmits it to the CM4000 through network and receives data from the CM400 through network and transmits it to the pseudo-tty port.

RS232 Property Configuration

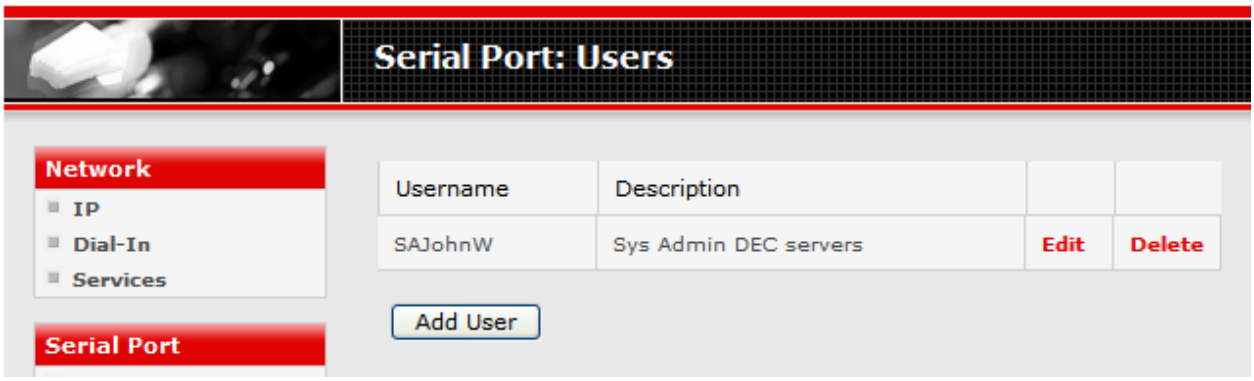
Once the protocols have been selected, you must now configure the serial port parameters for each port so they match the port parameters of the devices you have attached.

- Select the appropriate **Baud Rate**, **Parity**, **Data Bits**, **Stop Bits** and **Flow Control** for each port
- Click **Apply**

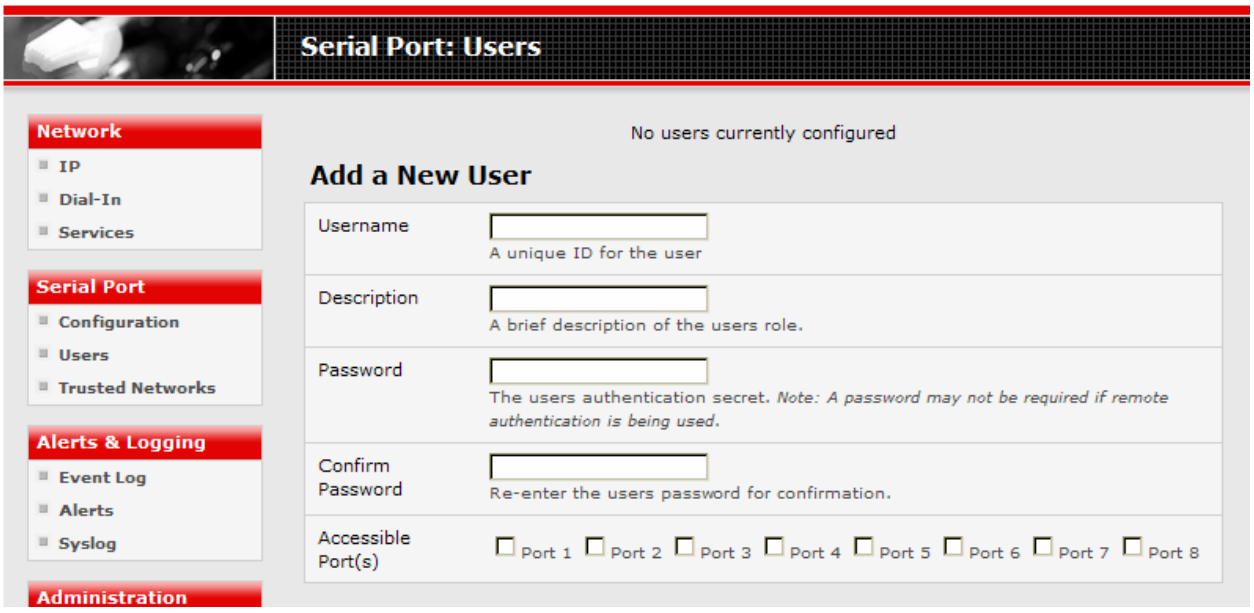
Note The serial ports are all set at the factory to 9600 baud, No parity, 8 data bits, 1 stop bit with software (Xon-Xoff) flow control enabled.

Add Users

This menu enables you to set up (and to delete) authenticated users and to define the port access permissions for each of these users:



- Select **Serial Port: Users**
- Click **Apply**

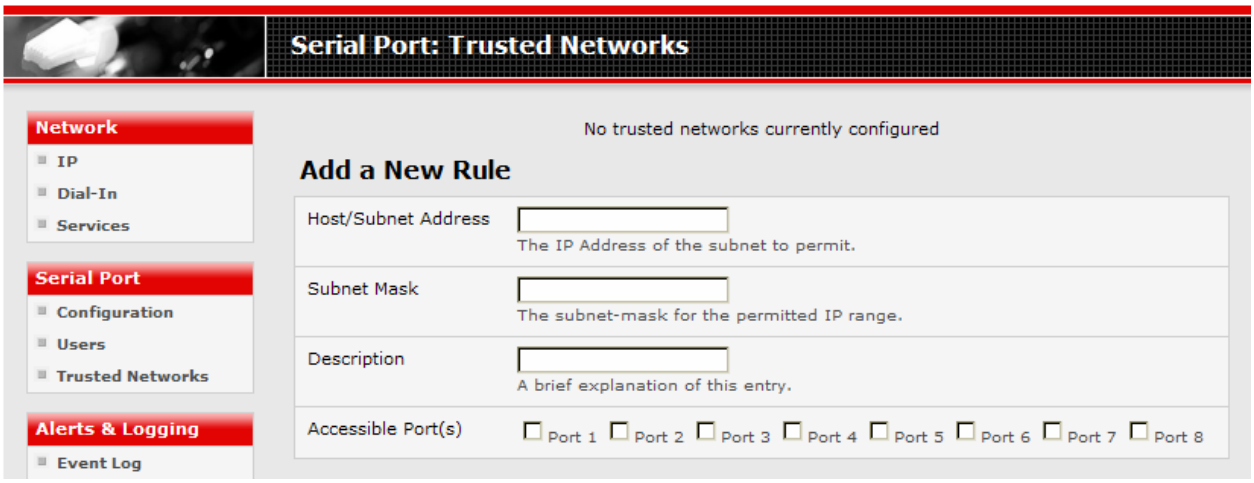


- Add a **Username** and **Password** for each new user. You may also include information related to the user (*e.g.* contact details) in the **Description** field
- You can now nominate which Ports you wish that user to have access to
- Click **Apply**

Your new user will now be able to access the serial device attached to the nominated serial port, locally or remotely. There are no limits on the number of users you can set up, or the number of users per port.

Trusted Networks

The **Trusted Networks** facility gives you an option to nominate specific IP addresses that Users must be located at, to have access to the CM4000 Ports.



- Select **Serial Port: Trusted Networks**
- To add a new trusted network, first enter the **Network IP Address** of the subnet to be permitted access.
- Then specify the range of addresses that are to be permitted by entering a **Subnet Mask** for that permitted IP range *e.g.*
  - To permit all the users located with a particular Class C network (204.15.5.0 say) connection to the nominated port then you would add the following Trusted Network New Rule:

Network IP Address	204.15.5.0
Subnet Mask	255.255.255.0

  - If you want to permit only the one users who is located at a specific IP address (204.15.5.13 say) to connect:

Network IP Address	204.15.5.0
Subnet Mask	255.255.255.255

- If however you want to allow all the users operating from within a specific range of IP addresses (say any of the thirty addresses from 204.15.5.129 to 204.15.5.158) to be permitted connection to the nominated port:

Host /Subnet Address	204.15.5.128
Subnet Mask	255.255.255.224

➢ Select the Port (or Ports) that the New Rule is to be applied to and click **Apply**

Note The above Trusted Networks will limit access by Users to the CM4000 serial ports. However they do not restrict access by the Administrator to the CM4000 console server itself. To change the default settings for this access, you will to need to edit the *IPtables* rules as described in the Chapter 11 - Advanced.

Chapter 5 Dial-In Access

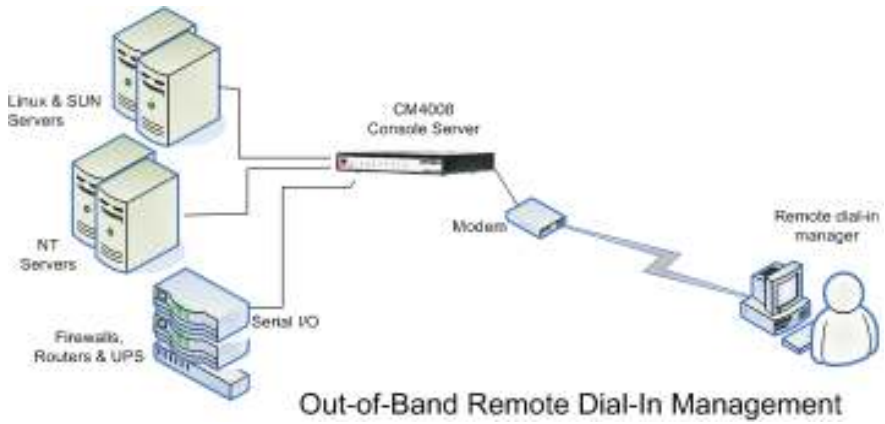
Introduction

The administrator can access the CM4000 out-of-band (OoB) from remote sites, using dial-up modem connections. There are three steps in setting up dial-in:

1. An external modem must be connected to the console server. The modem attaches *via* a serial cable to the DB9 console/modem port. The DB9 port is marked ***Local*** and is located on the back of the CM4008 unit, and the front of the CM4116/CM4148 unit

2. A dial-in PPP connection setting must be configured on the CM4000. Once configured for dial-in access, the CM4000 will then await incoming connection from a remote site

3. The PC (or workstation) at the remote site must configured with appropriate networking software to establish the dial-up PPP connection from the remote site to the CM4000. Once the PPP connection has established the remote user can then access the console server (using the browser or the command line interface) or connect to a console server serial port (via telnet, SSH or raw TCP/IP)

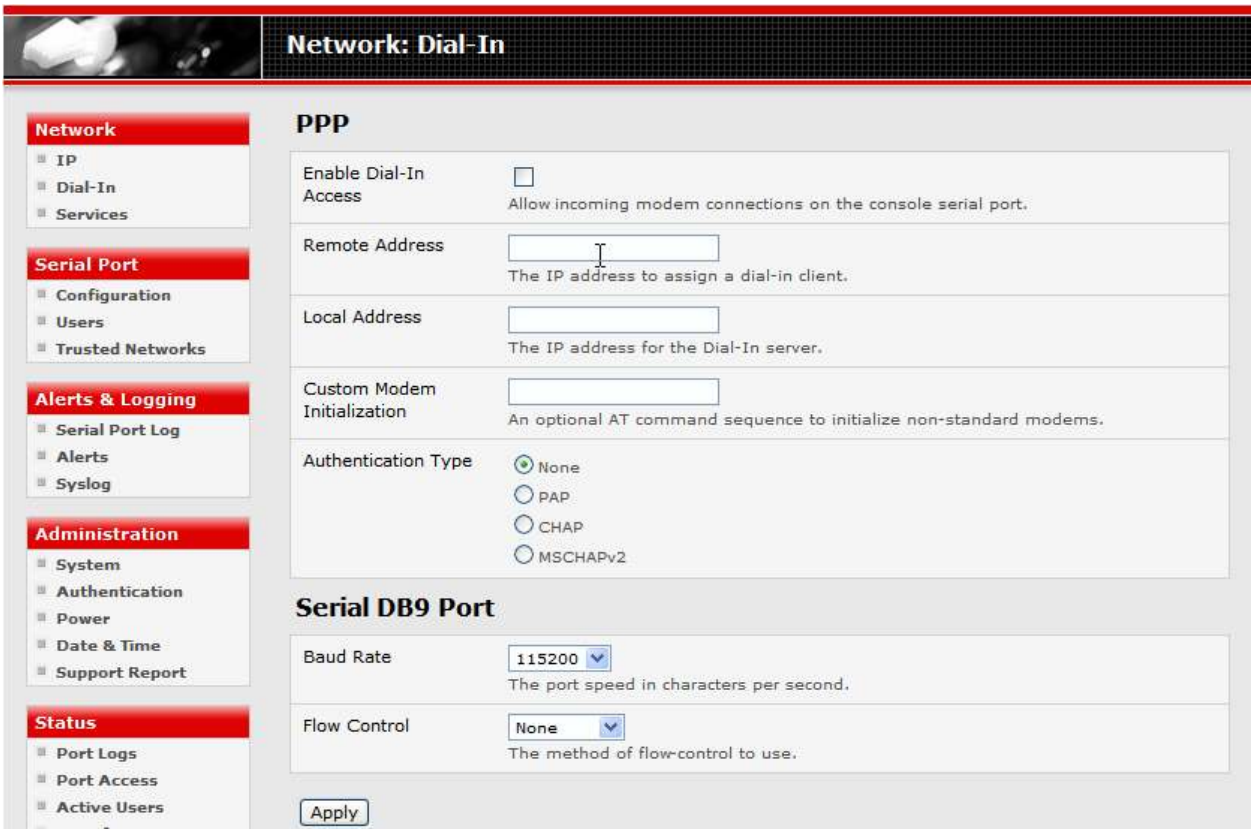


Out-of-Band Remote Dial-In Management

Configuring for dial-In PPP Access

To enable dial-in access to the console modem port:

- Select the **Network: Dial In** menu option



The console/modem port is set by default to 115200 baud, No parity, 8 data bits and 1 stop bit, with software (Xon-Xoff) flow control enabled. If this does not match the port settings for the modem, the modem/console port (or the modem) must be reconfigured:

- Under the **Serial DB9 Port** heading, select the **Baud Rate** and **Flow Control** that will communicate with the attached modem.

Note You can further configure the console/modem port (*e.g.* to include *modem init* strings) by editing */etc/mgetty.config* files as described in the Chapter 11 - Advanced.

To set up the PPP link, go to the **PPP** heading:

- Check the **Enable Dial-In Access** box
- In the **Remote Address** field, enter the IP address to be assigned to the dial in client. You can select any address for the Remote IP Address; but it, and the Local IP Address, must both be in the same network range (*e.g.* 200.100.1.12 and 200.100.1.67)
- In the **Local Address** field enter the IP address for the Dial-In PPP Server. This is the IP address that will be used by the remote client to access CM4000 once the modem connection is established. Again you can select any address for the Local IP Address but it must both be in the same network range as the Remote IP Address
- The **Custom Modem Initialization** option allows a custom AT string modem initialization string to be entered (*e.g*. AT&C1&D3&K3).
- Then you must select the **Authentication Type** to be applied to the dial-in connection. The CM4000 uses authentication to challenge administrators who dial in to the console server. The administrator must also have their client PC / workstation configured to use the selected authentication scheme. Select **PAP CHAP MSCHAPv2** or N**one** and click **Apply**

None With this selection, no username or password authentication is required for dial-in access. This is not recommended.

PAP Password Authentication Protocol (PAP) is the usual method of user authentication used on the internet: sending a username and password to a server where they are compared with a table of authorized users. Whilst most common, PAP is the least secure of the authentication options.

CHAP Challenge-Handshake Authentication Protocol (CHAP) is used to verify a user's name and password for PPP Internet connections. It is more secure than PAP, the other main authentication protocol.

MSCHAPv2 Microsoft Challenge Handshake Authentication Protocol (MSCHAP) is authentication for PPP connections between a computer using a Microsoft Windows operating system and a network access server. It is more secure than PAP or CHAP, and is the only option that also supports data encryption.

Database For dial-in access, the username and password received from the dial-in client are verified against the local authentication database stored on the CM4000.

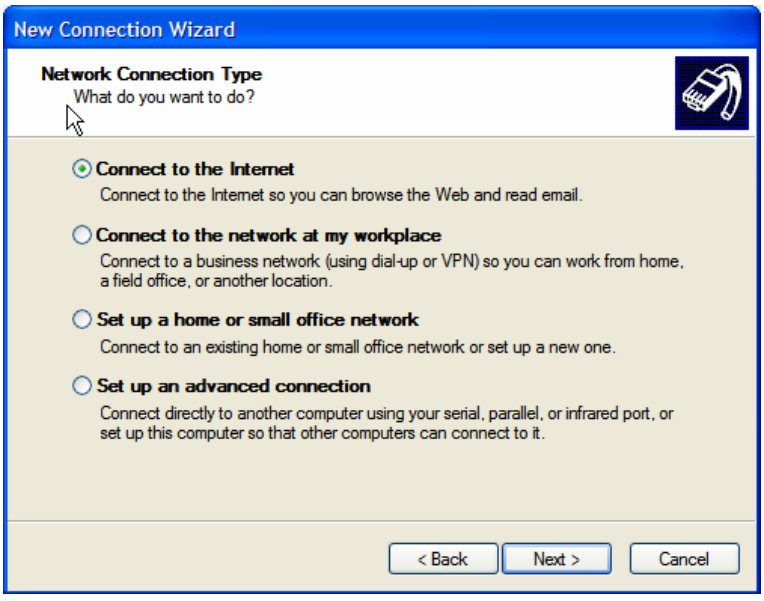
Note Chapter 11 (Advanced Configurations) has examples of Linux commands that can be used to control the modem port operation at the command line level

Set up the remote Client

For dial-in clients access you will need to set up a network connection from the client modem to the dial in modem on the remote CM4000:

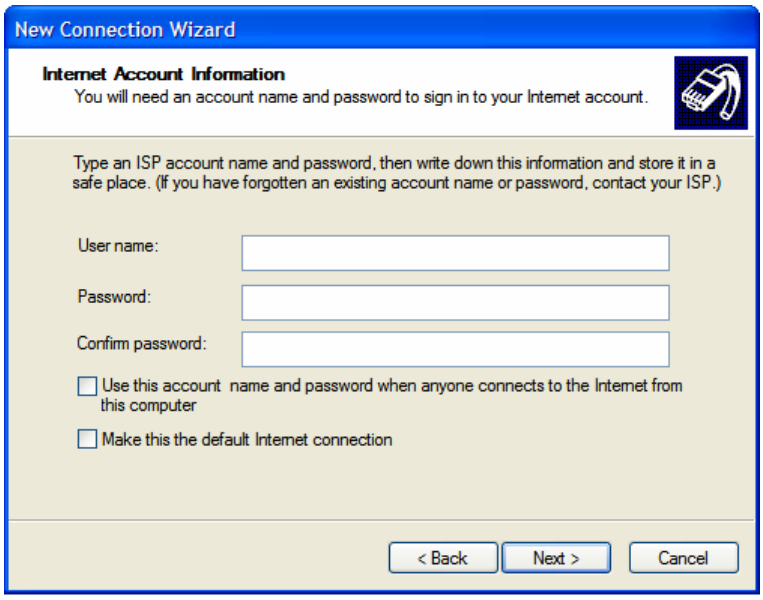
A. For Windows XP and Windows 2003 clients:

- Open **Network Connections** in Control Panel and click the **New Connection Wizard**


New Connection Wizard
Network Connection Type
What do you want to do?
Connect to the Internet
Connect to the Internet so you can browse the Web and read email.
Connect to the network at my workplace
Connect to a business network (using dial-up or VPN) so you can work from home, a field office, or another location.
Set up a home or small office network
Connect to an existing home or small office network or set up a new one.
Set up an advanced connection
Connect directly to another computer using your serial, parallel, or infrared port, or set up this computer so that other computers can connect to it.
< Back
Next >
Cancel

- Select **Connect to the Internet** and click **Next**
- On the **Getting Ready** screen select **Set up my connection manually** and click **Next**

- On the **Internet Connection** screen select **Connect using a dial-up modem** and click **Next**
- Enter a **Connection Name** (any name you choose) and the dial-up **Phone number** that will connect thru to the CM4000 modem


New Connection Wizard
Internet Account Information
You will need an account name and password to sign in to your Internet account.
Type an ISP account name and password, then write down this information and store it in a safe place. (If you have forgotten an existing account name or password, contact your ISP.)
User name:
Password:
Confirm password:
Use this account name and password when anyone connects to the Internet from this computer
Make this the default Internet connection
< Back
Next >
Cancel

- Enter a **User name** and **Password** for have set up for the CM4000

B. For clients running earlier Windows versions:

- For Windows 2000, the PPP client set up procedure is the same as above, except you get to the **Dial-Up Networking Folder** by clicking the **Start** button and selecting **Settings.** Then click **Network and Dial-up Connections** and click **Make New Connection**
- Similarly for Windows 98 you double click **My Computer** on the Desktop, then open **Dial-Up Networking** and double click **Make New Connection** and proceed as above

C. For Linux clients:

The online tutorial http://www.yolinux.com/TUTORIALS/LinuxTutorialPPP.html presents a selection of methods for establishing a dial up PPP connection:

- Command line PPP and manual configuration (which works with any Linux distribution)
- Using the Linuxconf configuration tool (for Red Hat compatible distributions). This configures the scripts *ifup/ifdown* to start and stop a PPP connection
- Using the Gnome control panel configuration tool -
- WVDIAL and the Redhat "Dialup configuration tool"
- GUI dial program X-isp. Download/Installation/Configuration

Note For all PPP clients:

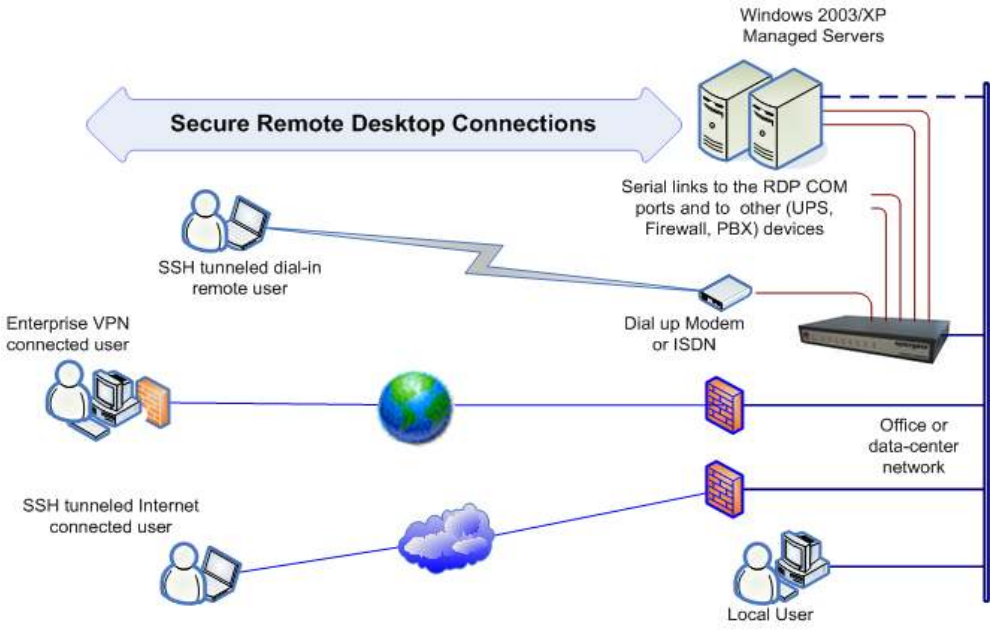
- Set the PPP link up with TCP/IP as the only protocol enabled
- Specify that the Server will assign IP address and do DNS
- Do not set up the CM4000 PPP link as the default for Internet connection

Chapter 6 Secure Desktop Tunneling

Introduction

Secure Desktop Tunneling is an easy to use remote management tool that allows both end users and administrators to securely access and take remote control of any computer (Windows, Linux, UNIX, Solaris, Macintosh) running Microsoft's Remote Desktop, or the popular VNC.

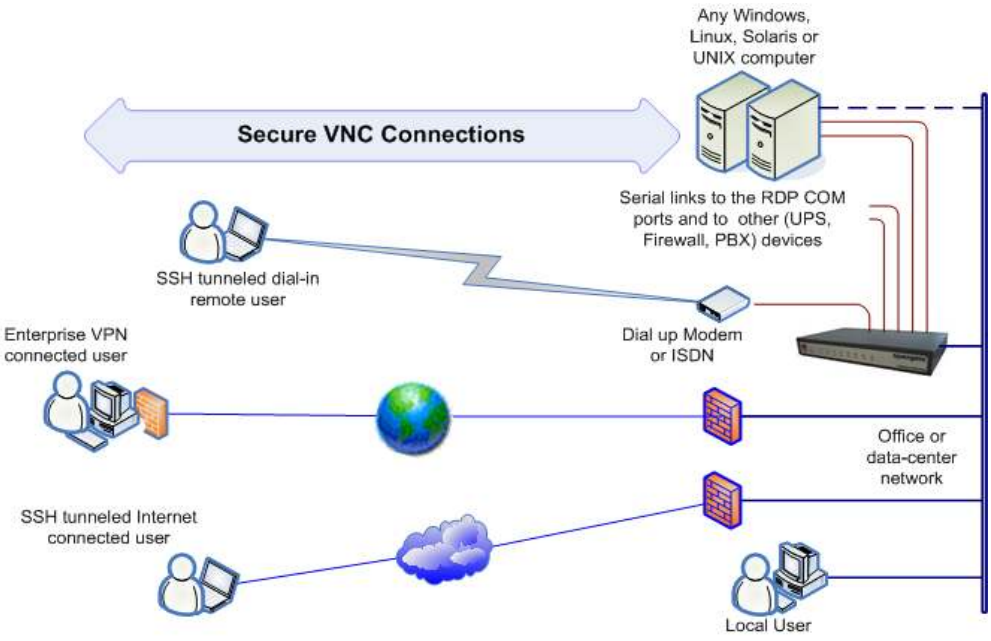
Microsoft's Remote Desktop Protocol (RDP) allows the remote user to connect to a work Windows XP or Windows 2003 computer and have access to all of the applications, files, and network resources (just as though they were in front of the computer screen at work). The remote administrator can also access and manage these computers (upgrade the server's operating system, reboot the machine etc). Opengear's Secure Desktop Tunneling implements SSH tunneling, so this RDP traffic is all securely transferred through an authenticated and encrypted tunnel.



Secure Remote Tunnels

Alternately, with Opengear's Secure Desktop Tunneling and Virtual Network Computing (VNC), users and administrators can securely access and control Windows 98/NT/2000/XP/2003, Linux, Macintosh, Solaris and UNIX computers. There's a range

of popular VNC software available (UltraVNC, RealVNC, TightVNC) - freely and commercially.


Any Windows, Linux, Solaris or UNIX computer
Secure VNC Connections
Serial links to the RDP COM ports and to other (UPS, Firewall, PBX) devices
SSH tunneled dial-in remote user
Dial up Modem or ISDN
Enterprise VPN connected user
Office or data-center network
SSH tunneled Internet connected user
Local User

Secure Remote VNC Tunnels

To set up Secure Desktop Tunnel access for RDP or VNC, the computer being accessed must be located on the same local network as the CM400, or cabled to the CM4000 via its serial COM port. The remote user/administrator then connects to the CM4000 via:

- a secure dial-up or ISDN modem (thru an SSH tunnel)
- a secure broadband Internet connection (thru an SSH tunnel)
- the enterprise VPN network (optionally thru an SSH tunnel) or
- the local network (optionally thru an SSH tunnel)

This chapter details:

- Setting up a Secure Desktop Tunnel for Remote Desktop (Section 6.1)
- Setting up a Secure Desktop Tunnel for VNC (Section 6.2)

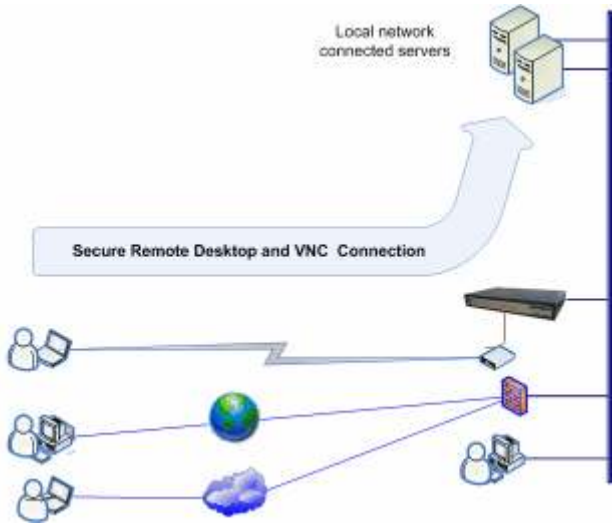
6.1 Secure Desktop Tunneling for Remote Desktop

The steps in setting up the Secure Remote Desktop connection are:

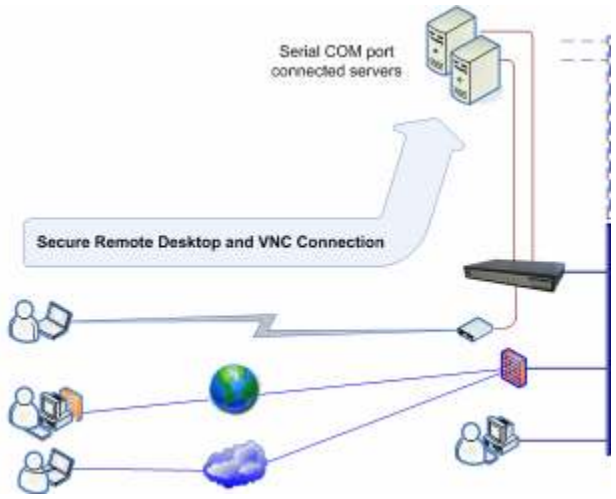
I. Enable Remote Desktop on the Windows computer that is to be accessed (Section 6.1.1)

II. Establish an RDP link from the CM4000 to the Windows computer:

A. For Windows computers that are network connected to the CM4000, you must set up **Secure Desktop Tunneling - Hosts** on the CM4000 (Section 6.1.2)



B. For Windows computers that are serially connected through their COM port to the CM4000, you must first establish a PPP connection (Section 6.1.3); then set up **Secure Desktop Tunneling - Ports** on the CM4000 (Section 6.1.4)



III. Establish the Remote Desktop connection between the Client PC and the CM4000 (Section 6.1.5):

A. For public or private network connected Clients you will need the public IP address of the CM4000, and to ensure Port 22 is forwarded through the network

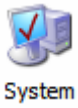
B. For dial-in Clients, you must first establish a PPP connection between the PC and the CM4000

IV. Then set up the secure SSH tunnel from Client PC to the CM4000. An SSH secure tunnel should be used for all public network connections (via dial-in or broadband Internet); and can be also be used for private network connections (local and enterprise VPN) (Section 6.1.6)

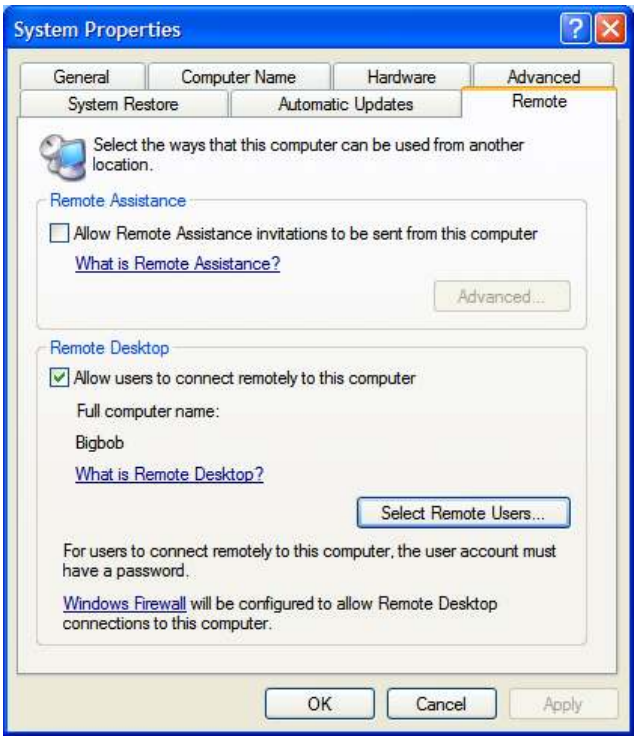
V. When the Client PC has been securely connected to the CM4000, then you can establish the Remote Desktop connection through to the Windows computer by simply configuring the RPD client software on the client PC (Section 6.1.7)

6.1.1 Enable Remote Desktop on the Windows computer to be accessed

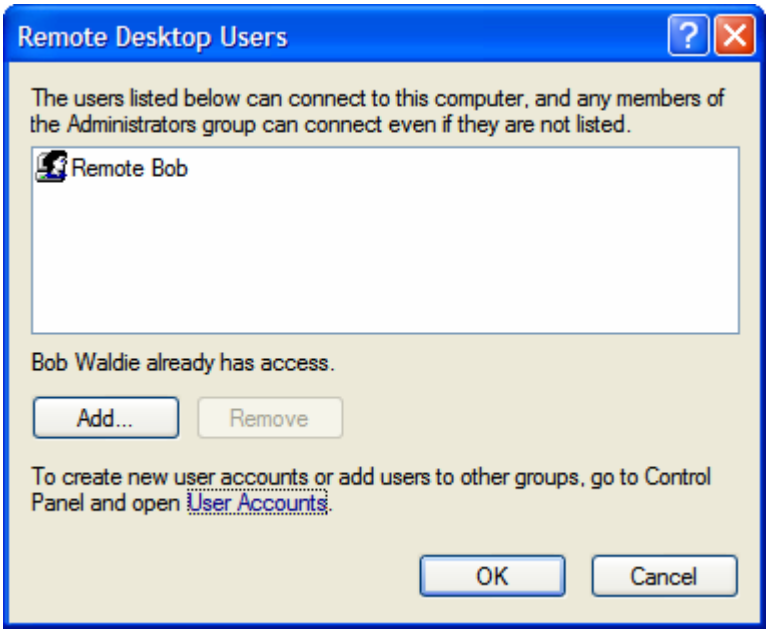
With Microsoft's Remote Desktop you can access and manage Windows XP Professional and Windows Server 2003 computers. To enable **Remote Desktop** on the Windows computer being accessed:



- Open **System** in the Control Panel and click the **Remote** tab



- Check **Allow users to connect remotely to this computer**
- Click **Select Remote Users**


Remote Desktop Users
The users listed below can connect to this computer, and any members of the Administrators group can connect even if they are not listed.
Remote Bob
Bob Waldie already has access.
Add...
Remove
To create new user accounts or add users to other groups, go to Control Panel and open User Accounts.
OK
Cancel

- To set the user(s) who can remotely access the system with RDP click **Add** on the **Remote Desktop Users** dialog box

Note If you need to set up new users for Remote Desktop access, open **User Accounts** in the Control Panel and proceed through the steps to nominate the new user's name, password and account type (Administrator or Limited)

Note With Windows XP Professional, you have only one Remote Desktop session and it connects directly to the Windows root console. With Windows Server 2003, you have the console session and two other general sessions - so more than one user can have active sessions on a single computer.

When the remote user connects to the accessed computer on the console session, Remote Desktop automatically locks that computer (so no other user can access the applications and files). When you come back to your computer at work, you can unlock it by typing CTRL+ALT+DEL.

6.1.2 Set up Secure Desktop Tunneling - Hosts on the CM4000

To set up *RDP (and VNC) forwarding* on the CM4000 to network connected computers:

- Select the **Network: SDT Hosts** menu option and click **Add Host**

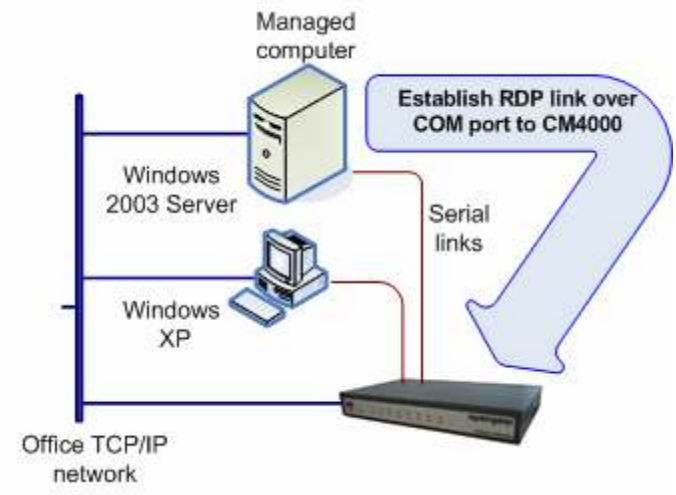
- Enter the **IP address/DNS Name** of the Windows computer (SDT Host) to be accessed
- Enter a **Description** (optional) for the Windows computer
- Select the **Permitted Users** who can have access to the Windows computer. You can add CM4000 Users (or reconfigure User profiles) by selecting **Serial Port:User** menu tag - as described earlier in Chapter 4 *Configuring Serial Ports*
- Click **Apply** This will enable RDP forwarding to the new host (and VNC forwarding, and SSH tunneling)

Note The following TCP Ports are used by SDT in the CM4000:

22	SSH (All Tunnelled connections)
3389	RDP on local LAN (forwarded inside tunnel)
5900	VNC on local LAN (forwarded inside tunnel)
73XX	RDP over serial from local LAN (forwarded inside tunnel) – where XX is the serial Port number
79XX	VNC over serial from local LAN (forwarded inside tunnel) – where XX is the serial Port number

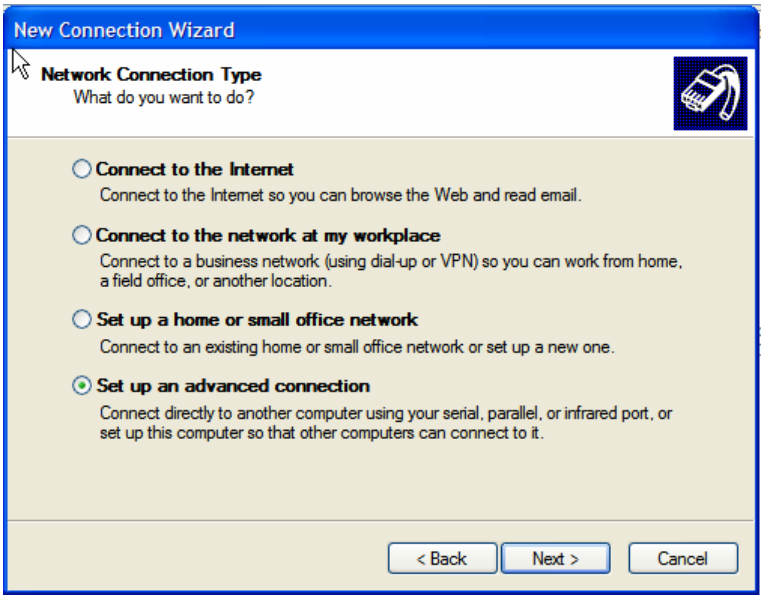
6.1.3 Establish a PPP connection from the computer's COM port to the CM4000 (only for serially connected computers)

Firstly, physically connect the COM port on the Windows computer that is to be accessed, to the serial port on the CM4000.

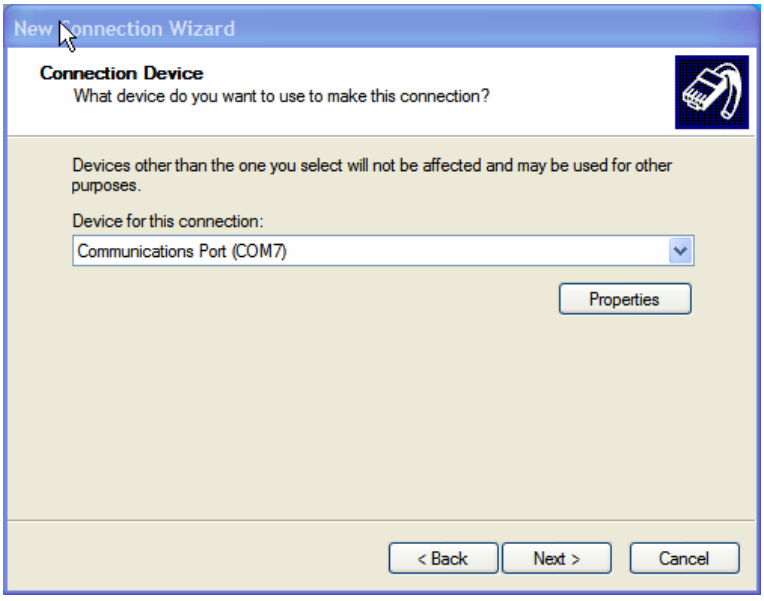


Then set up an advanced network connection between the Windows computer, through its COM port to the CM4000. Both Windows 2003 and Windows XP Professional allow you to create a *simple dial in service* which can be used for this Remote Desktop connection to the CM4000:

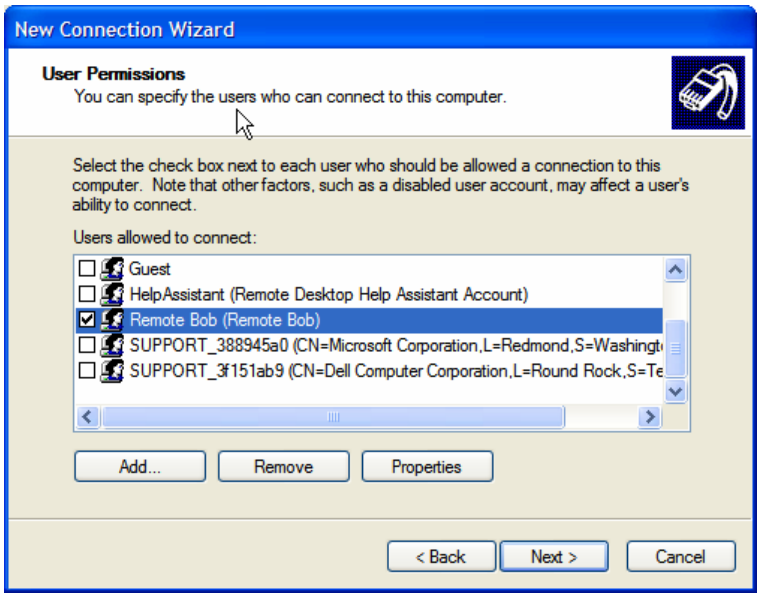
- Open **Network Connections** in Control Panel and click the **New Connection Wizard**


New Connection Wizard
Network Connection Type
What do you want to do?
Connect to the Internet
Connect to the Internet so you can browse the Web and read email.
Connect to the network at my workplace
Connect to a business network (using dial-up or VPN) so you can work from home, a field office, or another location.
Set up a home or small office network
Connect to an existing home or small office network or set up a new one.
Set up an advanced connection
Connect directly to another computer using your serial, parallel, or infrared port, or set up this computer so that other computers can connect to it.
< Back
Next >
Cancel

- Select **Set up an advanced connection** and click **Next**
- On the **Advanced Connection Options** screen select **Accept Incoming Connections** and click **Next**

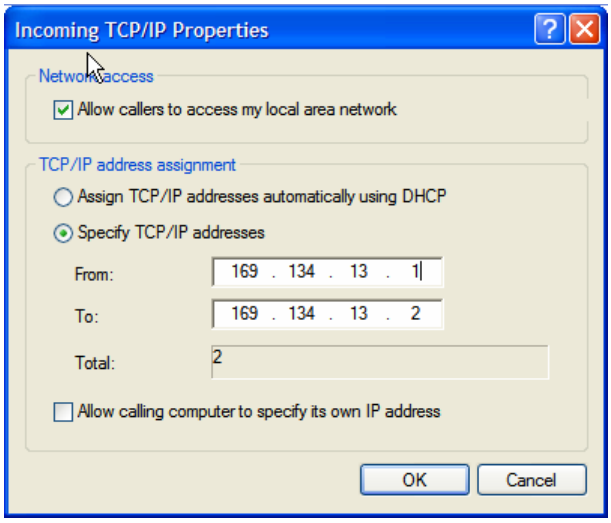

New Connection Wizard
Connection Device
What device do you want to use to make this connection?
Devices other than the one you select will not be affected and may be used for other purposes.
Device for this connection:
Communications Port (COM7)
Properties
< Back
Next >
Cancel

- Select the **Connection Device** (i.e. the serial COM port on the Windows computer that you cabled through to the CM4000). By default select **COM1**. The COM port on the Windows computer should be configured to its maximum baud rate. Click **Next**
- On the **Incoming VPN Connection Options** screen select **Do not allow virtual private connections** and click **Next**


New Connection Wizard
User Permissions
You can specify the users who can connect to this computer.
Select the check box next to each user who should be allowed a connection to this computer. Note that other factors, such as a disabled user account, may affect a user's ability to connect.
Users allowed to connect:
Guest
HelpAssistant (Remote Desktop Help Assistant Account)
Remote Bob (Remote Bob)
SUPPORT_388945a0 (CN=Microsoft Corporation,L=Redmond,S=Washingt
SUPPORT_3f151ab9 (CN=Dell Computer Corporation,L=Round Rock,S=Te
Add...
Remove
Properties
< Back
Next >
Cancel

- Specify which users will be allowed to use this connection. This should be the same users who were given Remote Desktop access privileges in the earlier step. Click **Next**

- On the **Network Connection** screen select **TCP/IP** and click **Properties**



- Select **Specify TCP/IP addresses** on the **Incoming TCP/IP Properties** screen select **TCP/IP.** Nominate a *From:* and a *To:* TCP/IP address and click **Next**

Note You can choose any TCP/IP addresses so long as they are addresses which are not used anywhere else on your network. The *From:* address will be assigned to the Windows XP/2003 computer and the *To:* address will be used by the CM4000.

For simplicity you can set the advanced connection and RDP access on the Windows computer to use the CM4000 defaults :

- Specify 10.233.111.254 as the *From:* address
- Select *Allow calling computer to specify its own address*

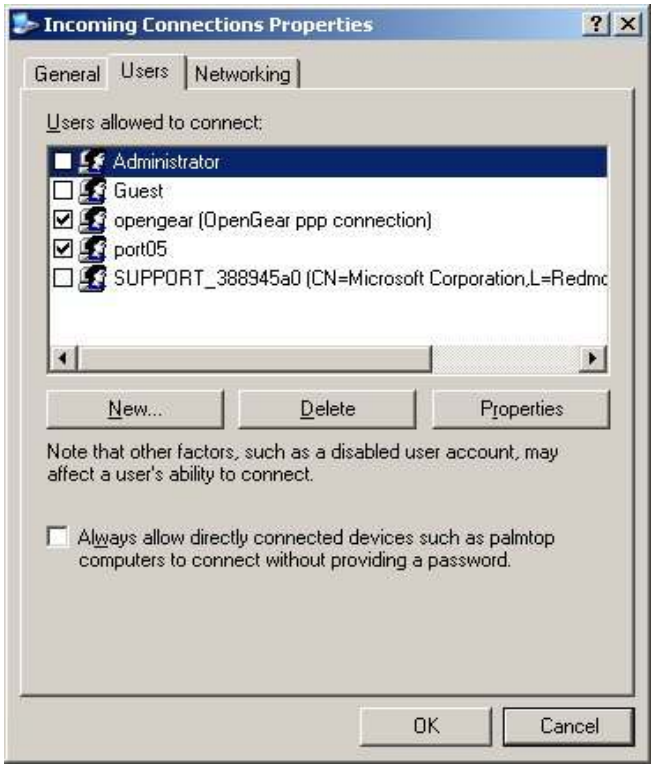
Also you could use the CM4000 default username and password when you set up the new Remote Desktop User and gave this User permission to use the advance connection to access the Windows computer:

- The CM4000 default *Username is portXX* where XX is the serial port number on the CM4000.
- The default *Password is portXX*

So to use the defaults for a RDP connection to the serial port 2 on the CM4000, you would have set up a Windows user named *port02*

- When the PPP connection has been set up, a network icon will appear in the Windows task bar

Note The above notes describe setting up an incoming connection for Windows XP. The steps are the same for Windows 2003, except that the set up screens present slightly differently:



Also the option for to **Set up an advanced connection** is not available in Windows 2003 if RRAS is configured. If RRAS has been configured it is a simply task to enable the null modem connection for the dial-in configuration.

6.1.4 Set up Secure Desktop Tunneling – Ports on the CM4000 (only for serially connected computers)

To set up *RDP (and VNC) forwarding* on the CM4000 Serial Port that is connected to the Windows computer COM port:

- Select the **Serial Port: Configuration** menu option and click **Edit** (for the particular Serial Port that is connected to the Windows computer COM port)

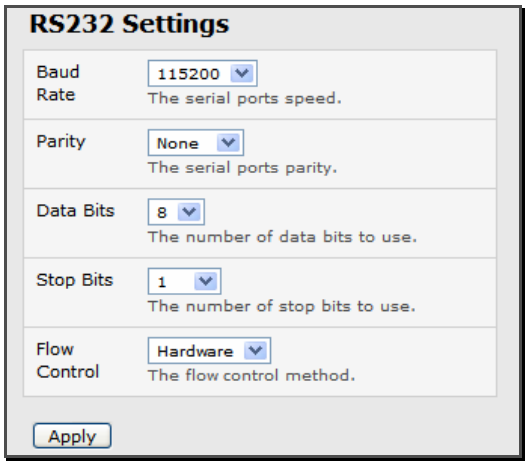
- Click **Secure Desktop Tunneling** This will enable RDP forwarding (and VNC forwarding, and SSH tunneling for these facilities)

> **Note** When you enable SDT, this will override all other Configuration protocols on that port

- Enter **Username** and **User Password** for the dial-in PPP link that was set up to the Windows computer

> **Note** If you leave the *Username* and *User Password* fields blank, they default to *portXX* and *portXX* where XX is the serial port number. So the default user name and password for Secure RDP over Port 2 is *port02*
>
> For the IP address fields, the default *Remote Address* is 10.233.111.254 and the default *Local Address* is 10.233.111.<portnumber> eg 10.233.111.2 for Secure RDP over Port 2

- Ensure the CM4000 **RS232 Settings** (Baud Rate, Flow Control) are the same as were set up on the Windows computer COM port and click **Apply**

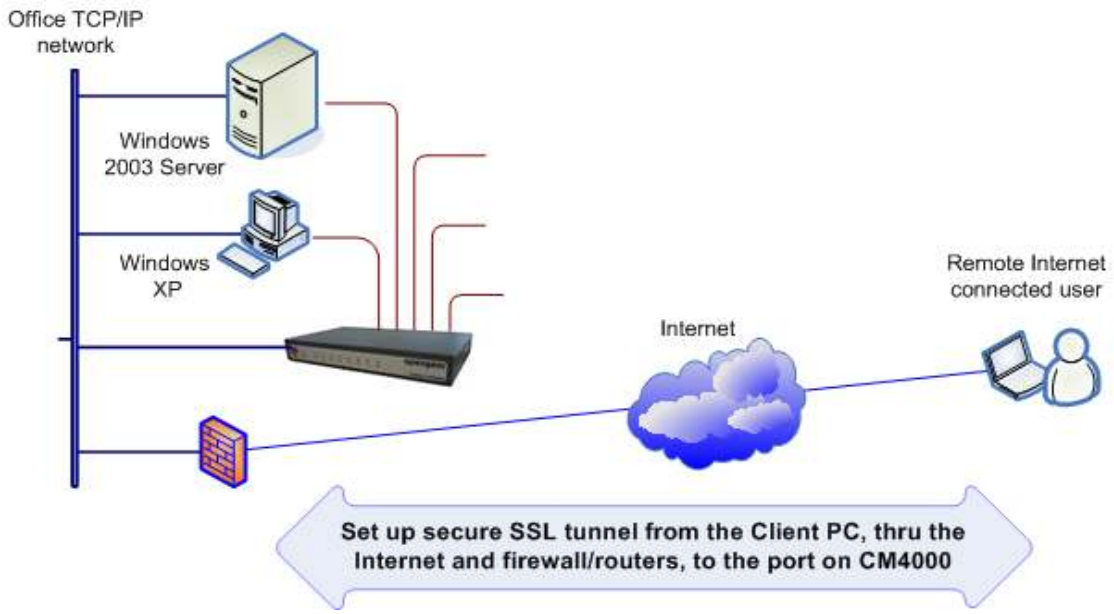


- RDP and VNC forwarding over serial ports is enabled on a Port basis. You can add Users who can have access to these ports (or reconfigure User profiles) by selecting **Serial Port:User** menu tag - as described earlier in Chapter 4 *Configuring Serial Ports*

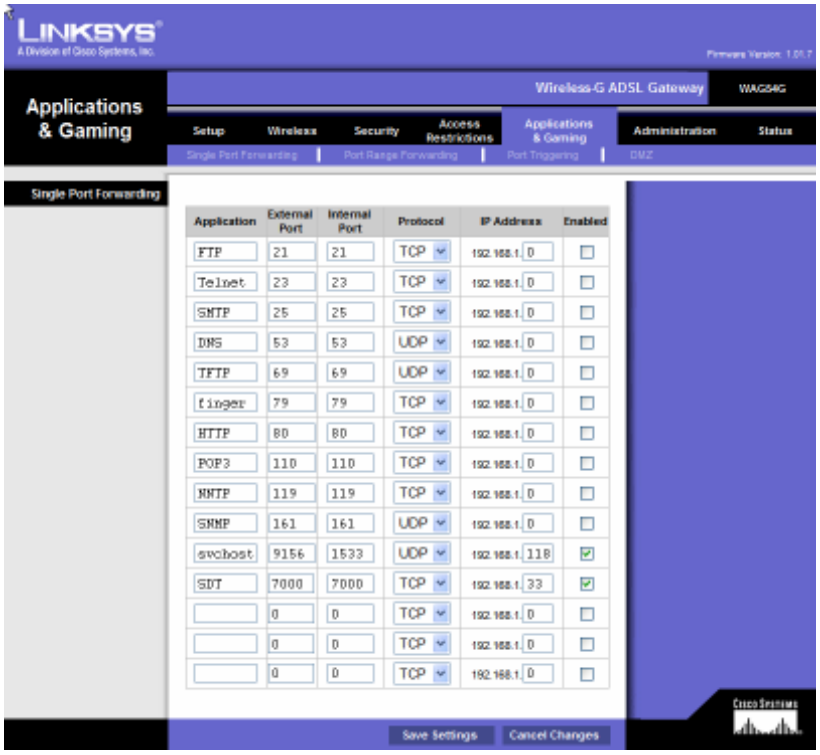
6.1.5 Establish a connection between the remote Client PC and the CM4000

A. If the remote RDP client PC is connecting to the CM4000 through the public Internet, before you can set up the secure SSH tunnel, you will need to:

- Determine *the public IP address* of the CM4000 (or of the router/firewall that connects the CM4000 to the Internet) as assigned by the ISP. To find the public IP address, access from http://checkip.dyndns.org/ or http://www.whatismyip.com/ from a PC on the same network as the CM4000 and note the reported IP address.

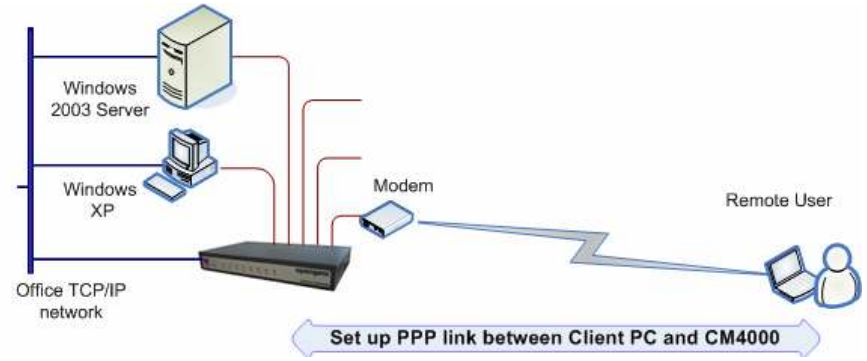

Office TCP/IP network
Windows 2003 Server
Windows XP
Internet
Remote Internet connected user
Set up secure SSL tunnel from the Client PC, thru the Internet and firewall/routers, to the port on CM4000

- Set port forwarding for TCP port 22 through any firewall/NAT/router that is located between the remote Client PC and the CM4000 e.g. the following shows the SDT SSH port being forwarded on a Cisco/Linksys WAG54G DSL gateway so it points to port 22 on the CM4000 that is located at 192.168.1.33



Note http://www.portforward.com has port forwarding instructions for a range of routers. Also you can use the Open Port Check tool from http://www.canyouseeme.org to check if port forwarding through local firewall/NAT/router devices has been properly configured

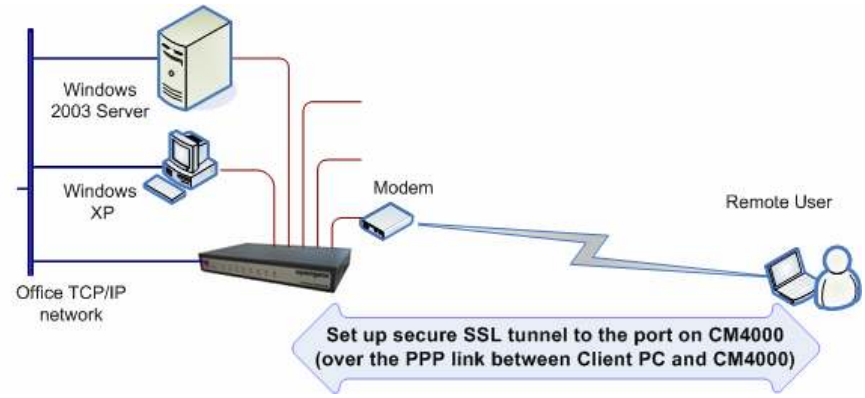
B. If the RDP client PC is dialing into *Local/Console* port on the CM4000 you will need to set up a dial-in PPP link:



- Configure the CM4000 for dial-in access (following the steps in the ***Configuring for Dial-In PPP Access*** section in Chapter 5, *Configuring Dial In Access*)

- Set up the PPP client software at the remote user PC (following the ***Set up the remote Client*** section in Chapter 5)

Once you have a dial-in PPP connection established, you then can set up the secure SSH tunnel from the remote Client PC to the CM4000.

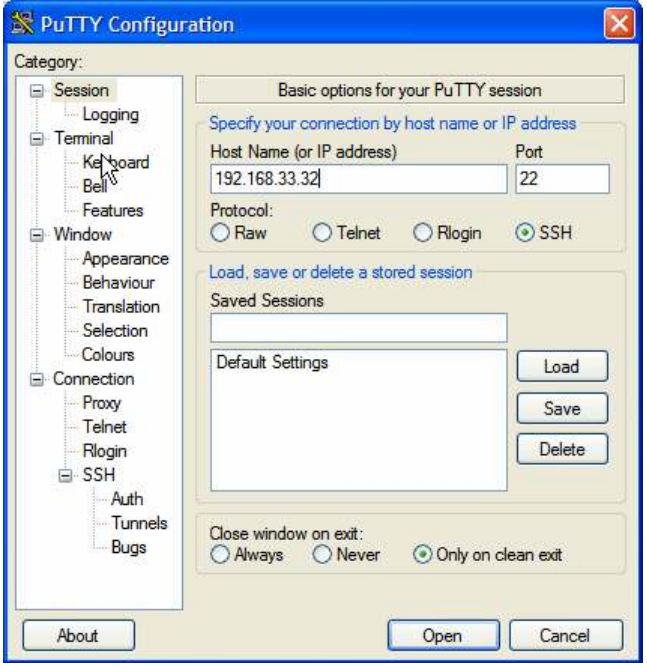


6.1.6 Create the SSH tunnel

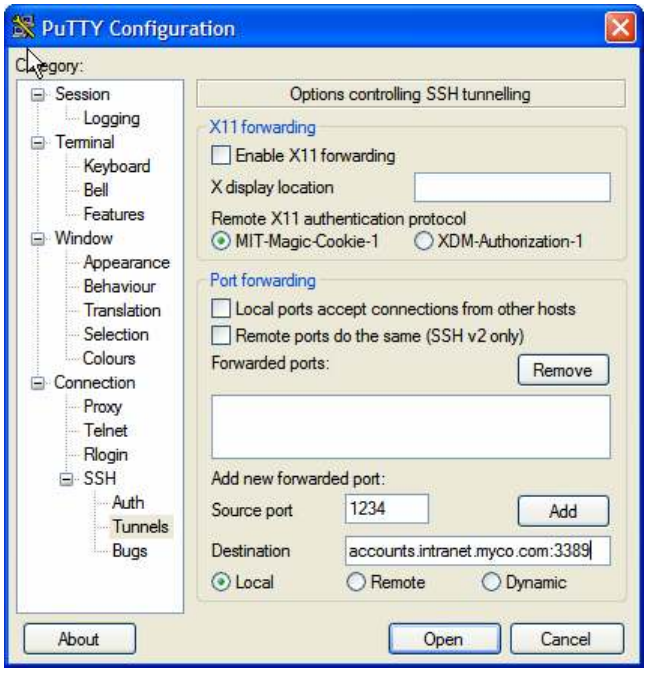
To set up the secure SSH tunnel from the remote Client PC to the CM4000, you must install and launch SSH client software on the remote Client PC. There's a wide selection of commercial and free SSH client programs available:

- PuTTY is a complete (though not very user friendly:) freeware implementation of SSH for Win32 and UNIX platforms
- SSHTerm is a useful open source SSH communications package
- SSH Tectia is leading end-to-end commercial communications security solution for the enterprise
- Reflection for Secure IT (formerly F-Secure SSH) is another good commercial SSH-based security solution

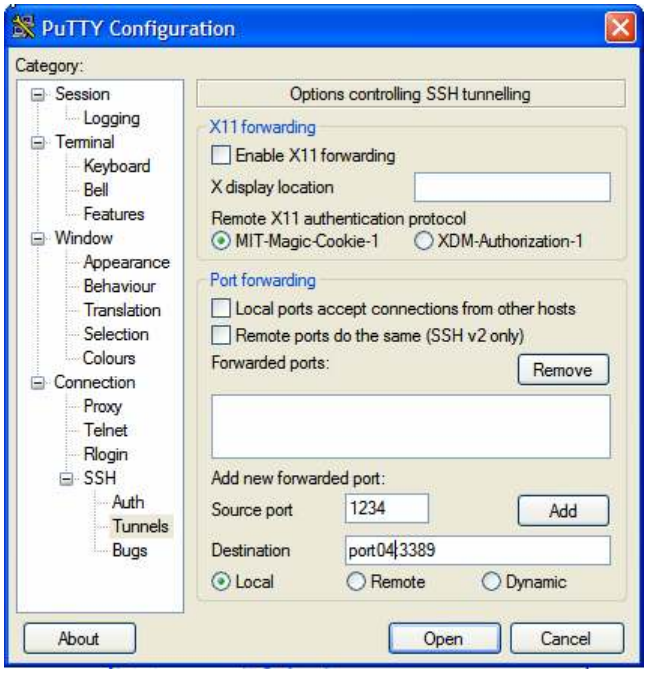
The steps below show the establishment of an SSH connection and then forwarding the RDP port over this SSH connection - using the PuTTY client software:



- Under the Session tab, enter the IP address of the CM4000 in the **Host Name or IP address** field.
  - For dial-in connections, this IP address will be the **Local** Address that you assigned to the CM4000 when you set it up as the Dial-In PPP Server
  - For Internet (or local/VPN connections) connections this will be the public IP address of the CM4000
- Select the **SSH Protocol**, and the **Port** will be set as 22
- Under the **SSH -> Tunnels** tab, **Add new forwarded port** specifying the **Source port** as *1234* (or any number you choose)
- Set the **Destination:**
  - If your destination computer is network connected to the CM4000, set the Destination as <SDT Host IP address/DNS Name>*:3389* e.g. if the SDT Host IP Address you specified when setting up the *SDT Hosts* on the CM4000 was *accounts.myco.intranet.com*, then specify the Destination as *accounts.myco.intranet.com:3389*



- If your destination computer is serially connected to the CM4000, set the Destination as *<port label>:3389* e.g. if the **Label** you specified on the SDT enabled serial port on the CM4000 is *win2k3*, then specify the remote host as *win2k3:3389* . Alternative you can set the Destination as *portXX:3389* where XX is the SDT enabled serial port number e.g. if port 4 is on the CM4000 is to carry the RDP traffic then specify *port04:3389*



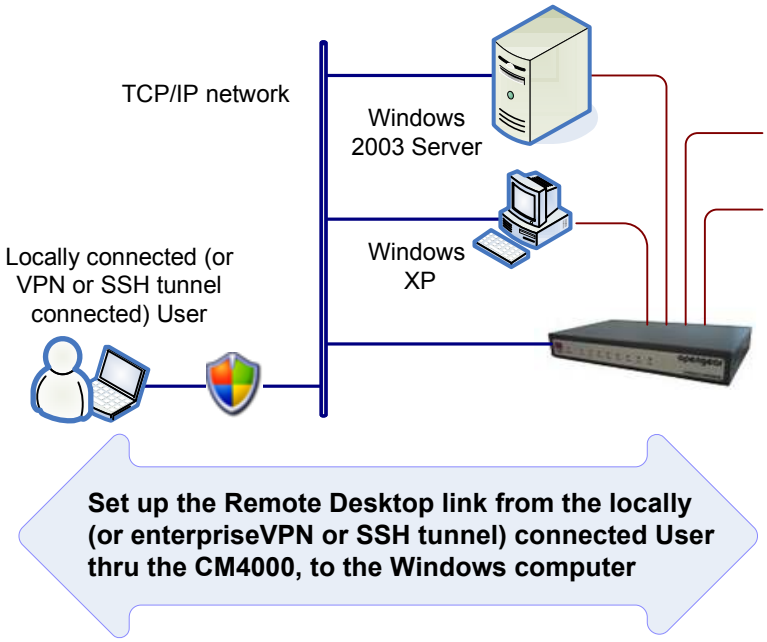
- Select **Local** and click the **Add** button
- Click **Open** to SSH connect the Client PC to the CM4000
- You will now be prompted for the Username/Password for the CM400 user you SDT enabled

Note You can also secure the RDP communications from local and enterprise VPN connected Client PCs using SSH as above.

This will protect against the risk of the “man in the middle” attacks to which RDP has a vulnerability http://www.securiteam.com/windowsntfocus/5EP010KG0G.html

6.1.7 Configure the Remote Desktop Connection client

Now you have the Client PC securely connected to the CM4000 (either locally, or remotely - thru the enterprise VPN, or a secure SSH internet tunnel or a dial-in SSH tunnel) you can establish the Remote Desktop connection from the Client.


TCP/IP network
Windows 2003 Server
Windows XP
Locally connected (or VPN or SSH tunnel connected) User
Set up the Remote Desktop link from the locally (or enterpriseVPN or SSH tunnel) connected User thru the CM4000, to the Windows computer

To do this connection you simply enable the **Remote Desktop Connection** on the remote client PC then point it to the Secure Desktop Tunnel port in the CM4000:

A. On a Windows client PC:

- Click **Start**. Point to **Programs**, then to **Accessories**, then **Communications**, and click **Remote Desktop Connection**

- In **Computer**, enter the appropriate IP Address and Port Number:
  - Where there is a direct local or enterprise VPN connection, enter the IP Address of the CM4000, and the Port Number of the Secure Desktop Tunnel for the CM4000 serial port that is attached to the Windows computer to be controlled e.g. if the Windows computer is connected to serial Port 3 on a CM4000 located at 192.168.0.50 then you would enter *192.168.0.50:7303*
  - Where there is an SSH tunnel (over a dial up PPP connection or over a public internet connection or private network connection) simply enter the *localhost* as the IP address i.e. *127.0.0.1* For Port Number, enter the *source port* you created when setting SSH tunneling /port forwarding (in Section 6.1.6) eg *:1234*
- Click **Option.** In the **Display** section specify an appropriate color depth (eg for a modem connection it is recommended you not use over 256 colors). In **Local Resources** specify the peripherals on the remote Windows computer that are to be controlled (printer, serial port etc)

- Click **Connect**

Note The Remote Desktop Connection software is pre-installed with Windows XP, however for earlier Windows PCs you will need to download the RDP client:

- Go to the Microsoft Download Center site http://www.microsoft.com/downloads/details.aspx?familyid=80111F21-D48D-426E-96C2-08AA2BD23A49&displaylang=en and click the **Download** button

  This software package will install the client portion of Remote Desktop on Windows 95, Windows 98 and 98 Second Edition, Windows Me, Windows NT 4.0, Windows 2000, and Windows 2003. When run, this software allows these older Windows platforms to remotely connect to a computer running Windows XP Professional or Windows 2003 Server

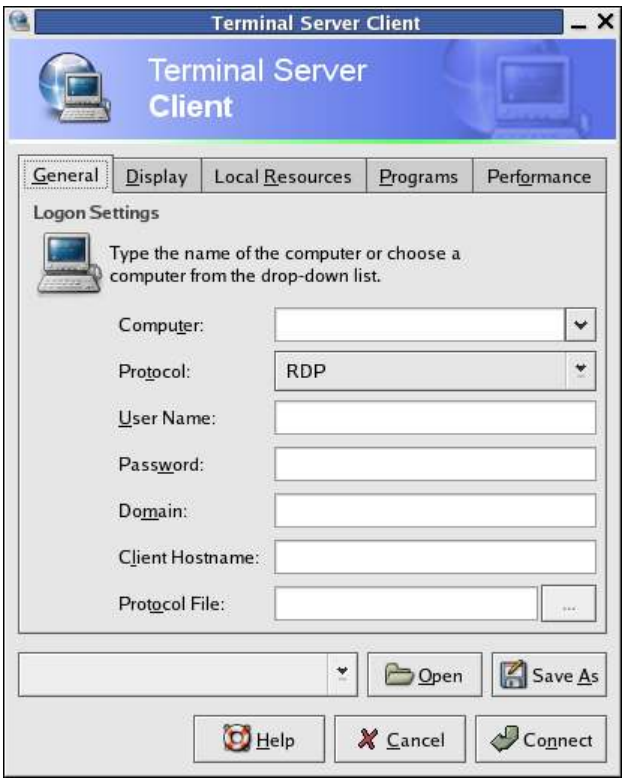
B. On a Linux or UNIX client PC:

➢ Launch the open source *rdesktop* client:

rdesktop -u *windows-user-id* -p *windows-password* -g 1200x950 *ms-windows-terminal-server-host-name*

option	description
-a	Color depth: 8, 16, 24
-r	Device redirection. i.e. Redirect sound on remote machine to local device i.e. -0 -r sound (MS/Windows 2003)
-g	Geometry: *widthxheight* or 70% screen percentage.
-p	Use -p - to receive password prompt.

➢ You can use GUI front end tools like the GNOME Terminal Services Client *tsclient* to configure and launch the *rdesktop* client. (Using *tsclient* also enables you to store multiple configurations of *rdesktop* for connection to many servers)



Note The *rdesktop* client is supplied with Red Hat 9.0:

- rpm -ivh rdesktop-1.2.0-1.i386.rpm

For Red Hat 8.0 or other distributions of Linux; download source, untar, configure, make, make then install.

rdesktop currently runs on most UNIX based platforms with the X Window System and can be downloaded from http://www.rdesktop.org/

C. On a Macintosh client:

➢ Download Microsoft's free Remote Desktop Connection client for Mac OS X http://www.microsoft.com/mac/otherproducts/otherproducts.aspx?pid=remotedesktopclient

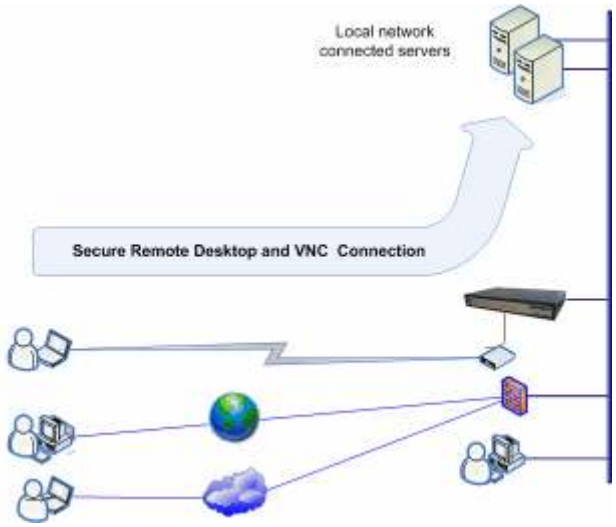
6.2 Secure Desktop Tunnel for VNC

The steps in setting up the secure VNC connection are:

I. Install and configure the VNC Server software on the computer to be accessed (Section 6.2.1)

II. Establish a network connect from the CM4000 to the computer:

A. For computers that are network connected to the CM4000, you must set up **Secure Desktop Tunneling - Hosts** on the CM4000 (Section 6.2.2)



B. For computers that are serially connected through their COM port to the CM4000, you must first establish a PPP connection (Section 6.2.3); then set up **Secure Desktop Tunneling - Ports** on the CM4000 (Section 6.2.4)

III. Establish a connection between the Viewer PC and the CM4000 (Section 6.2.5):

VI. Then set up the secure SSH tunnel from Viewer PC to the CM4000 (Section 6.2.6)

VII. Install and configure the VNC Viewer software on the Viewer PC (Section 6.2.7)

6.2.1 Install and configure the VNC Server on the computer to be accessed

Virtual Network Computing (VNC) software enables users to remotely access computers running Linux, Macintosh, Solaris, UNIX, all versions of Windows and most other operating systems.

A. For Microsoft Windows servers (and clients):

Windows does not include VNC software, so you will need to download, install and activate a third party VNC Server software package:

RealVNC http://www.realvnc.com is fully cross-platform, so a desktop running on a Linux machine may be displayed on a Windows PC, on a Solaris machine, or on any number of other architectures. There is a Windows server, allowing you to view the desktop of a remote Windows machine on any of these platforms using exactly the same viewer. RealVNC was founded by members of the AT&T team who originally developed VNC.

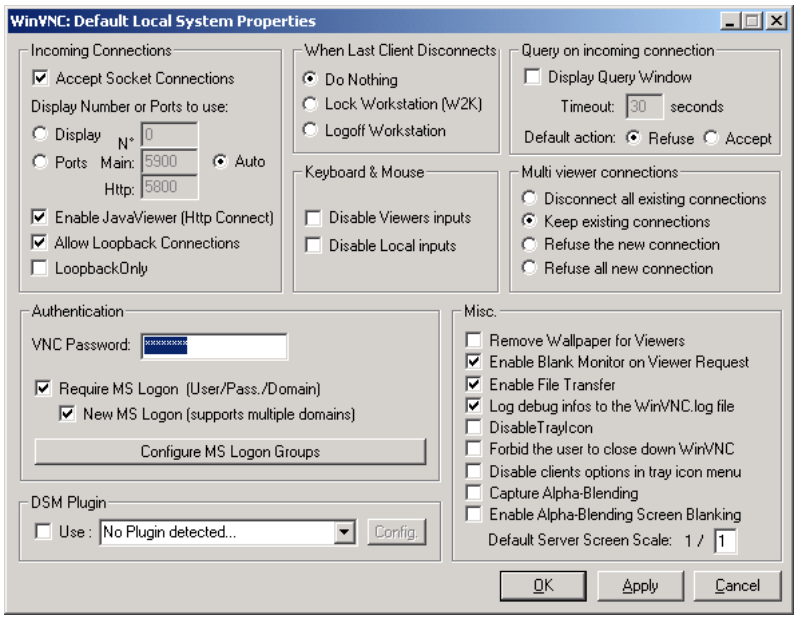
TightVNC http://www.tightvnc.com is an enhanced version of VNC. It has added features such as file transfer, performance improvements, and read-only password support. They have just recently included a video drive much like UltraVNC's. TightVNC is still free, cross-platform (Windows Unix and Linux) and compatible with the standard (Real) VNC.

UltraVNC http://ultravnc.com is easy to use, fast and free VNC software that has pioneered and perfected features that the other flavors have consistently refused or been very slow to implement for cross platform and minimalist reasons. UltraVNC runs under Windows operating systems (95, 98, Me, NT4, 2000, XP, 2003) Download UltraVNC from Sourceforge's UltraVNC file list

So, for example, to install and configure the UltraVNC Server on Windows computer, you first select a language (e.g. English) then use the **Set Up** wizard to install the Server software:

Setup - UltraVNC
Welcome to the UltraVNC Setup Wizard
This will install UltraVNC v1.0 RC20.7 on your computer.
It is recommended that you close all other applications before continuing.
Click Next to continue, or Cancel to exit Setup.
UltraVNC
Next >
Cancel

Configuring the UltraVNC Server Refer is equally straightforward (though you should refer to http://doc.uvnc.com for more detailed Server (and Viewer) instructions)



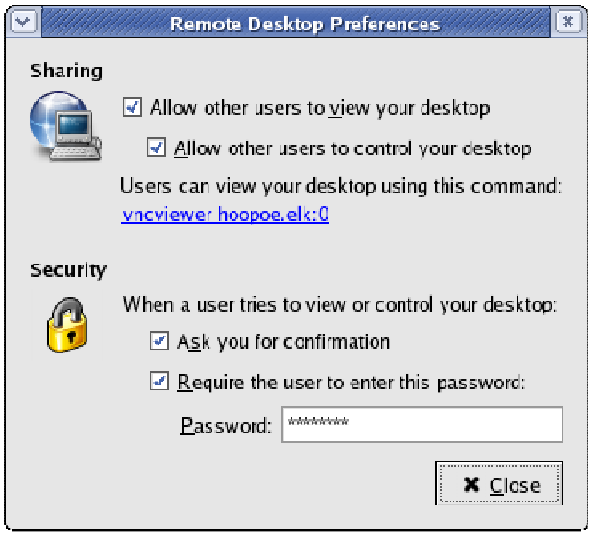
B. For Linux servers (and clients):

Most Linux distributions now include VNC Servers and Viewers and they are generally can be launched from the (Gnome/KDE etc) front end

e.g. with Red Hat Enterprise Linux 4 there's VNC Server software and a choice of Viewer client software, and to launch:

- Select the **Remote Desktop** entry in the **Main Menu -> Preferences** menu

- Click the **Allow other users…** checkbox to allow remote users to view and control your desktop


Remote Desktop Preferences
Sharing
Allow other users to view your desktop
Allow other users to control your desktop
Users can view your desktop using this command:
vncviewer hoopoe.elk:0
Security
When a user tries to view or control your desktop:
Ask you for confirmation
Require the user to enter this password:
Password:
Close

- To set up a persistent VNC server on Red Hat Enterprise Linux 4:
  - Set a password using **vncpasswd**
  - Edit **/etc/sysconfig/vncservers**
  - Enable the service with **chkconfig vncserver on**
  - Start the service with **service vncserver start**
  - Edit **/home/*username*/.vnc/xstartup** if you want a more advanced session than just *twm* and an xterm

C. For Macintosh servers (and clients):

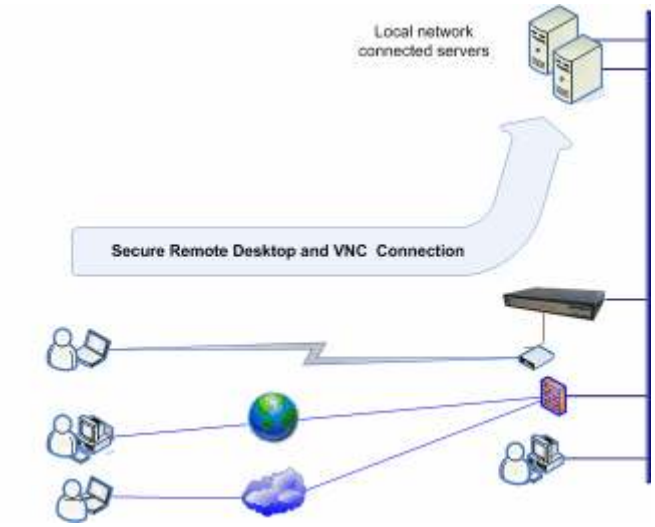
OSXvnc http://www.redstonesoftware.com/vnc.html is a robust, full-featured VNC server for Mac OS X that allows any VNC client to remotely view and/or control the Mac OS X machine. OSXvnc is supported by Redstone Software

D. Most other operating systems (Solaris, HPUX, PalmOS etc) either come with VNC bundled, or have third party VNC software that you can download.

6.2.2 Set up Secure Desktop Tunneling - Hosts on the CM4000

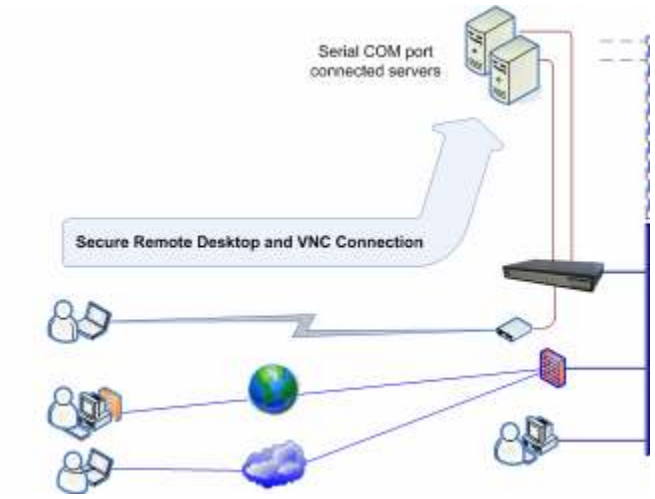
For computers that are network connected to the CM4000, you must set up *RDP (and VNC) forwarding* on the CM4000:

- Follow the steps in Section 6.1.2



6.2.3 Establish a PPP connection from the computer's COM port to the CM4000 (only for serially connected computers)

For computers that are serially connected from their COM port to the serial port on the CM4000, you must establish the PPP network connection, and then set up Secure Desktop Tunneling - Ports on the CM4000



- To establish the PPP network connection between the serial ports:

A. For non Windows (Linux, UNIX, Solaris etc) computers establish a PPP connection over the serial port. The online tutorial http://www.yolinux.com/TUTORIALS/LinuxTutorialPPP.html presents a selection of methods for establishing a PPP connection for Linux

B. For Windows XP and 2003 computers follow the steps in Section 6.2.3

C. For earlier version Windows computers again follow t he steps in Section 6.2.3, however to get to the **Make New Connection** button:

   - For Windows 2000, click **Start** and select **Settings** then at the **Dial-Up Networking Folder** click **Network and Dial-up Connections** and click **Make New Connection.** Note you may need to first set up connection over the COM port using **Connect directly to another computer** before proceeding to **Set up an advanced connection**
   - For Windows 98 you double click **My Computer** on the Desktop, then open **Dial-Up Networking** and double click

6.2.4 Set up Secure Desktop Tunneling – Ports on the CM4000 (only for serially connected computers)

For computers that are serially connected to the CM4000, you must set up *RDP (and VNC) forwarding* on the CM4000:

- To set up the above, follow the steps in Section 6.1.4

6.2.5 Establish a connection between the Viewer PC and the CM4000

For a remote Viewer PC you must establish a secure connection between the Viewer PC and the CM4000:

A. When the remote Viewer PC is dialing-in to the CM4000, you must first establish a PPP link

B. When the remote Viewer PC is connecting to the CM4000 via a public Internet (or private LAN) connection, you must ensure that TCP Port 22 is forwarded through all the firewall/NAT/routers

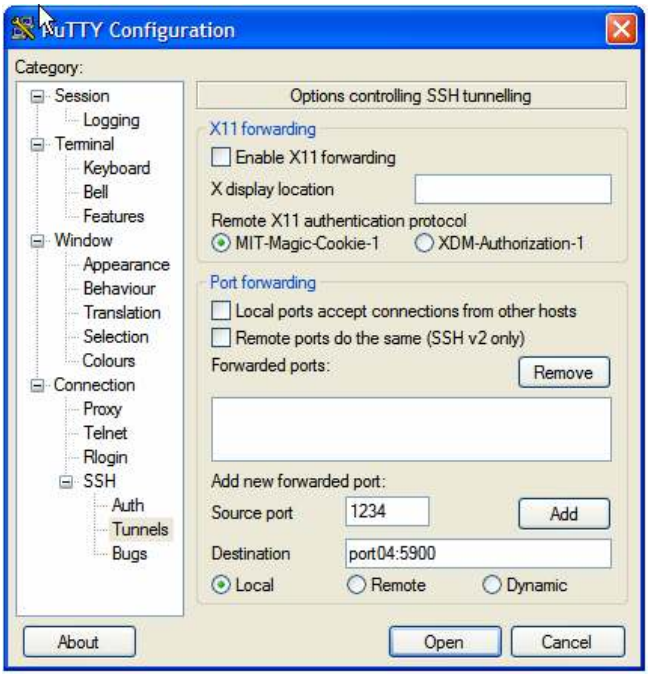
- To set up the above, follow the steps in Section 6.1.5

6.2.6 Create the SSH tunnel

To set up the secure SSH tunnel from the Viewer PC to the CM4000:

- Follow the steps in Section 6.1.6 - however when configuring the VNC port redirection specify port 5900 (rather than port 3389 as was used for RDP)

e.g. if using PuTTY:



Note How secure is VNC?

VNC access generally allows access to your whole computer, so security is very important. VNC uses a random challenge-response system to provide the basic authentication that allows you to connect to a VNC server. This is reasonably secure and the password is not sent over the network.

However, once connected, all subsequent VNC traffic is unencrypted. So a malicious user could snoop your VNC session. Also there are VNC scanning programs available, which will scan a subnet looking for PCs which are listening on one of the ports which VNC uses.

Tunneling VNC over a SSH connection ensures all traffic is strongly encrypted. Also no VNC port is ever open to the internet, so anyone scanning for open VNC ports will not be able to find your computers. When tunnelling VNC over a SSH connection, the only port which you're opening on your CM4000 the SDT port 22.

So sometimes it may be prudent to tunnel VNC through SSH even when the Viewer PC and the CM4000 are both on the same local network.

6.2.7 Install, configure and connect the VNC Viewer

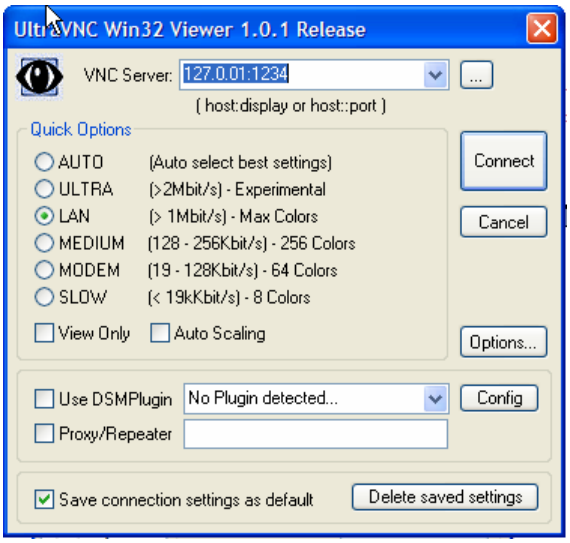
VNC is truly *platform-independent* so a VNC Viewer on any operating system can connect to a VNC Server on any other operating system. There are Viewers (and Servers) from a wide selection of sources (e.g. UltraVNC TightVNC or RealVNC) for most operating systems. There are also a wealth of Java viewers available so that any desktop can be viewed with any Java-capable browser (http://en.wikipedia.org/wiki/VNC lists many of the VNC Viewers sources).

- Install the VNC Viewer software and set it up for the appropriate speed connection

> Note To make VNC faster, when you set up the Viewer:
>
> - Set encoding to ZRLE (if you have a fast enough CPU)
> - Decrease color level (eg 64 bit)
> - Disable the background transmission on the Server or use a plain wallpaper
>
> (Refer to http://doc.uvnc.com for detailed configuration instructions)

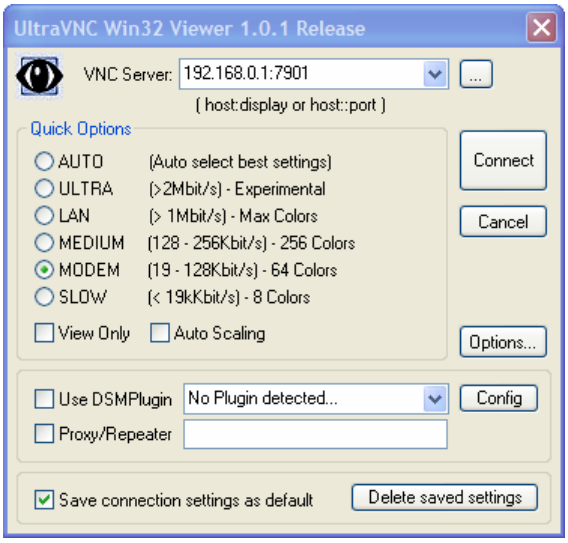
- To establish the VNC connection, first configure the VNC Viewer, entering the VNC Server IP address

A. When the Viewer PC is connected to the CM4000 thru a SSH tunnel (over a dial-in connection, or a public or private network connection), enter *locahost* (or 127.0.0.1) as the IP VNC Server IP address; and *the source port* you entered when setting SSH tunneling /port forwarding (in Section 6.2.6) eg *:1234*



B. When the Viewer PC is connected directly to the CM4000 (i.e. locally or remotely through a VPN); and the VNC Host computer is serially connected to the CM400; enter the IP address of the CM4000 unit with the TCP port that the SDT tunnel will use. The TCP port will be 7900 plus the physical serial port number (*i.e.* 7901 to 7948, so all traffic directed to port 79xx on the CM4000 is tunneled thru to port 5900 on the PPP connection on serial Port xx)

   e.g. for a Windows Viewer PC using UltraVNC connecting to a VNC Server which is attached to Port 1 on a CM4000 located 192.168.0.1



- You can then establish the VNC connection by simply activating the VNC Viewer software on the Viewer PC and entering the password

Note For general background reading on Remote Desktop and VNC access we recommend the following:

- *The Microsoft Remote Desktop How-To* http://www.microsoft.com/windowsxp/using/mobility/getstarted/remoteintro.mspx
- *The Illustrated Network Remote Desktop* help page http://theillustratednetwork.mvps.org/RemoteDesktop/RemoteDesktopSetupandTroubleshooting.html
- *What is Remote Desktop in Windows XP and Windows Server 2003?* by Daniel Petri http://www.petri.co.il/what's_remote_desktop.htm
- *Frequently Asked Questions about Remote Desktop* http://www.microsoft.com/windowsxp/using/mobility/rdfaq.mspx
- *Secure remote access of a home network using SSH, Remote Desktop and VNC for the home user* http://theillustratednetwork.mvps.org/RemoteDesktop/SSH-RDP-VNC/RemoteDesktopVNCandSSH.html
- *Taking your desktop virtual with VNC,* RedHat magazine http://www.redhat.com/magazine/006apr05/features/vnc/ and http://www.redhat.com/magazine/007may05/features/vnc/
- *Wikipedia* general background on VNC http://en.wikipedia.org/wiki/VNC

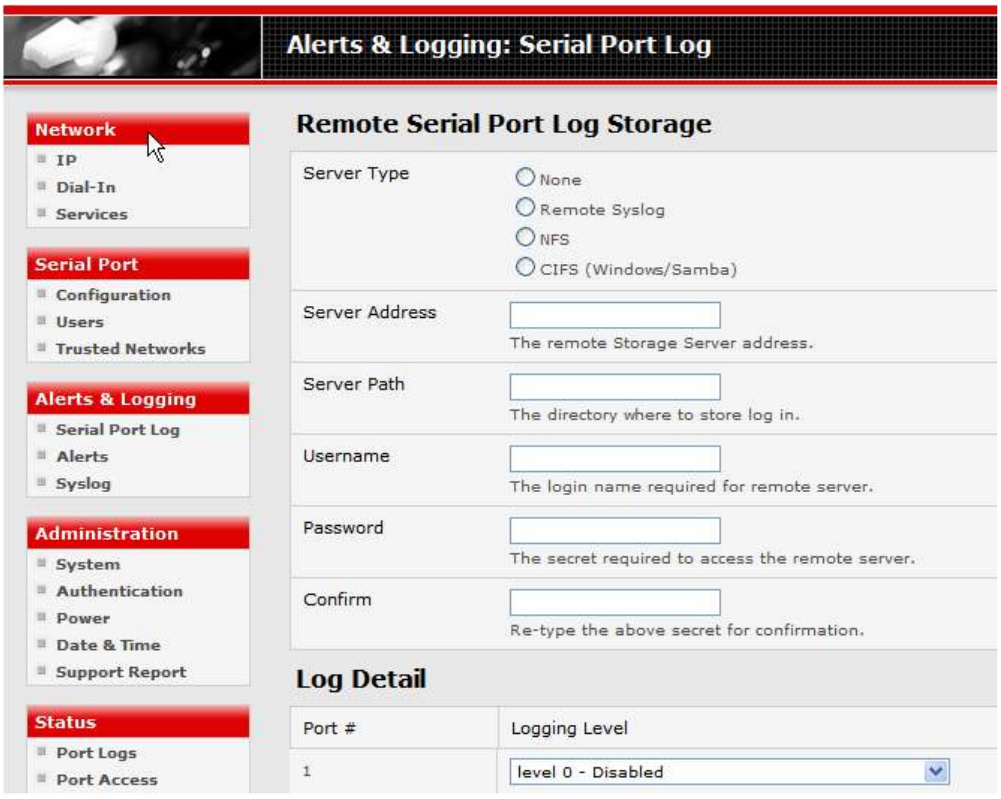
Chapter 7 Alerts and Logging

Introduction

This chapter describes the logging and alert generation features of the console server:

- The CM4000 serial port event logging allows Administrators to maintain a record of all communications with the servers and devices they are controlling
- The Alert facility monitors the ports and emails alerts when specified activity events occur
- A log of all system activity is also maintained

Serial Port Logging



Logging keeps an off-server record of all port activity for each nominated port.

To activate **Serial Port Logging** you must specify which ports are to have activities logged (and to what level) and where those logs are to be saved/forwarded to:

- Select the **Alerts & Logging: Serial Port Log** menu option
- Specify the **Server Type** to be used (and the details to enable log server access)
- Specify the **Logging Level** of for each port.

  0 Turns off logging for the selected port

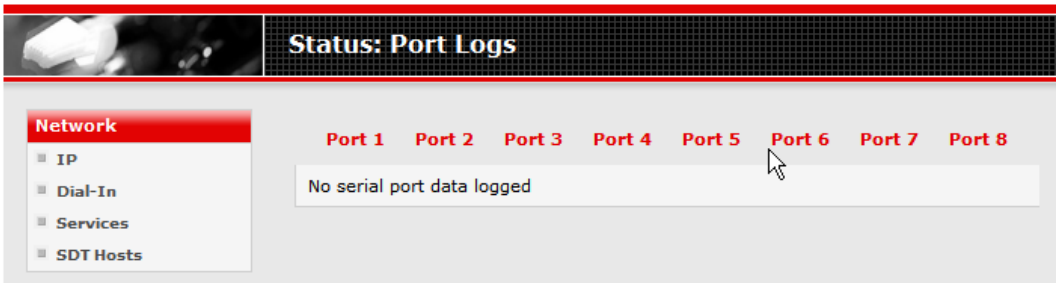
  1 Logs all connection events to the port

  2 Logs all data transferred to and from the port and all changes in hardware flow control status and all user connection events

- Click **Apply**. As you apply your event logging selections, the screen will be updated with a confirmation message:

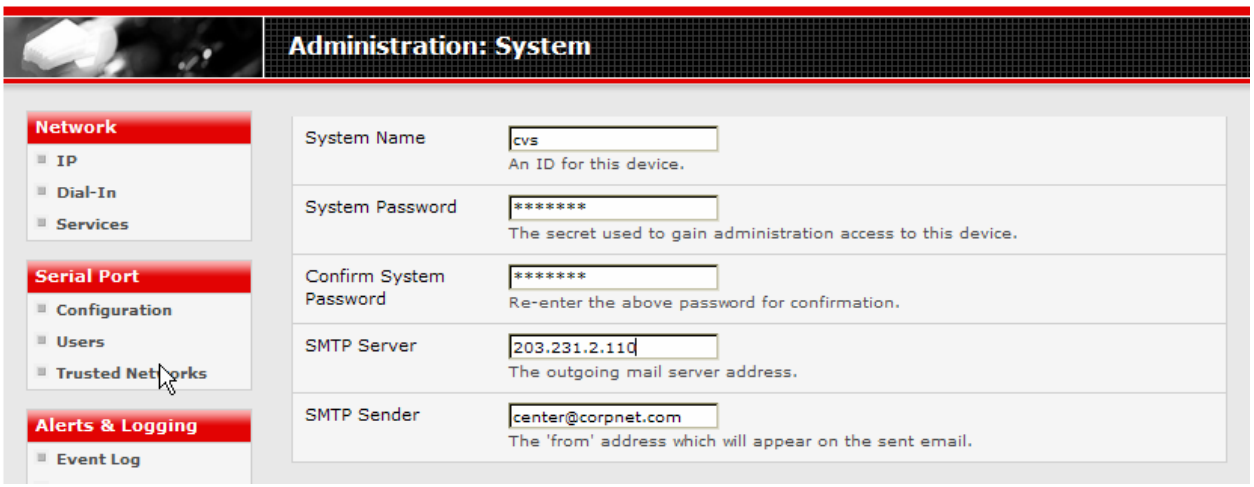
  Message Changes to configuration succeeded.

- To view the logged serial port data select **Status : Port Logs**



Email Alerts

With the Alerts facility enabled, the data stream from the nominated port is monitored for trigger conditions. When triggered, an alert message is emailed to a nominated email address.

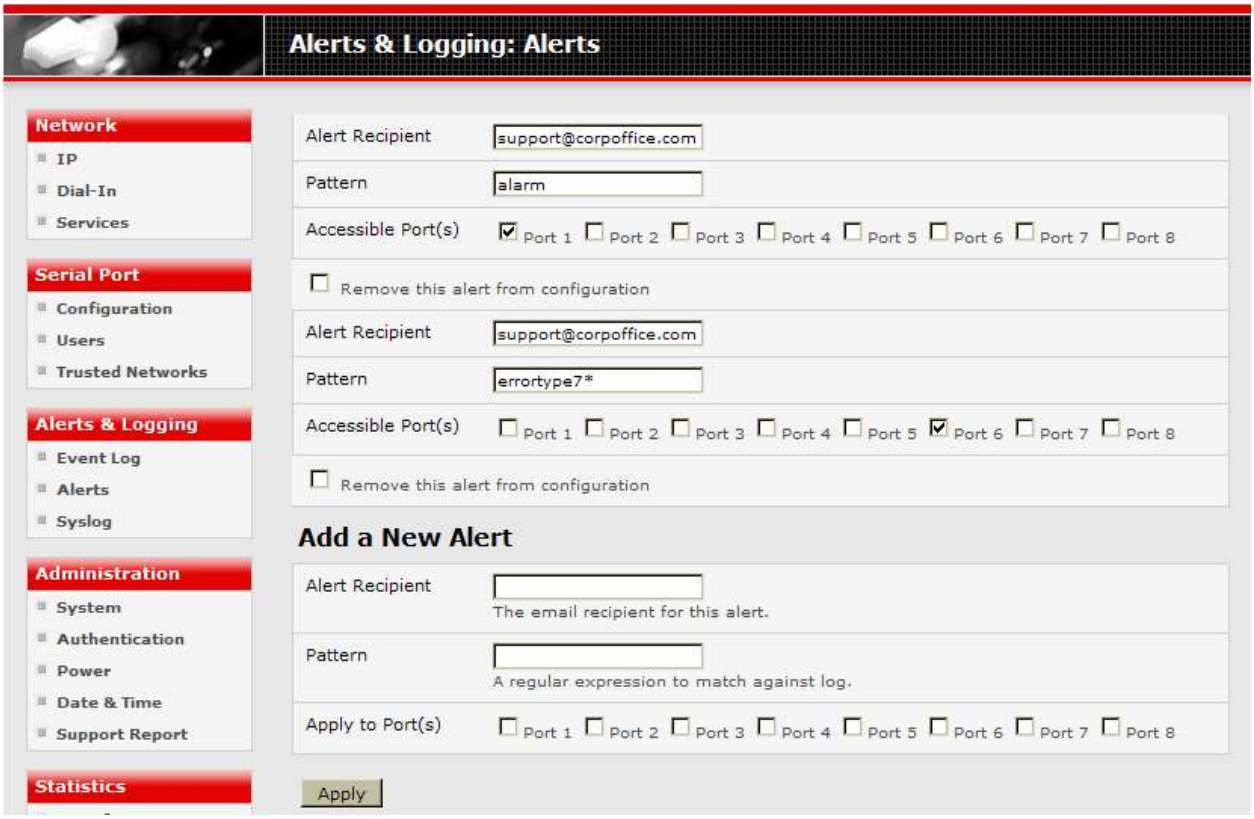


Before setting up the alert trigger, you must to setup email:

- Select **Administration: System** and specify the IP address of the outgoing SMTP Mail server
- You may also wish to enter an **SMTP Senders** email address which will appear as the from address in all sent email from this CM4000
- Click **Apply** to activate SMTP

- Select **Alerts & Logging: Alerts** and specify the email address for the Alert Recipient

- Set the *Pattern* to be scanned for (to trigger the alert) and which ports are to be scanned. Click **Apply**

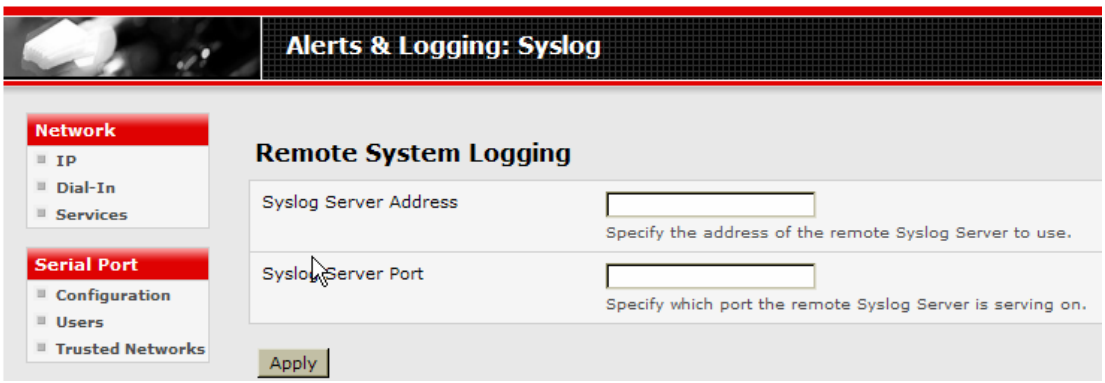


> **Note** Chapter 11 (Advanced Configurations) discusses using Linux commands scripts to create custom alert triggers and custom responses to alerts

Syslog

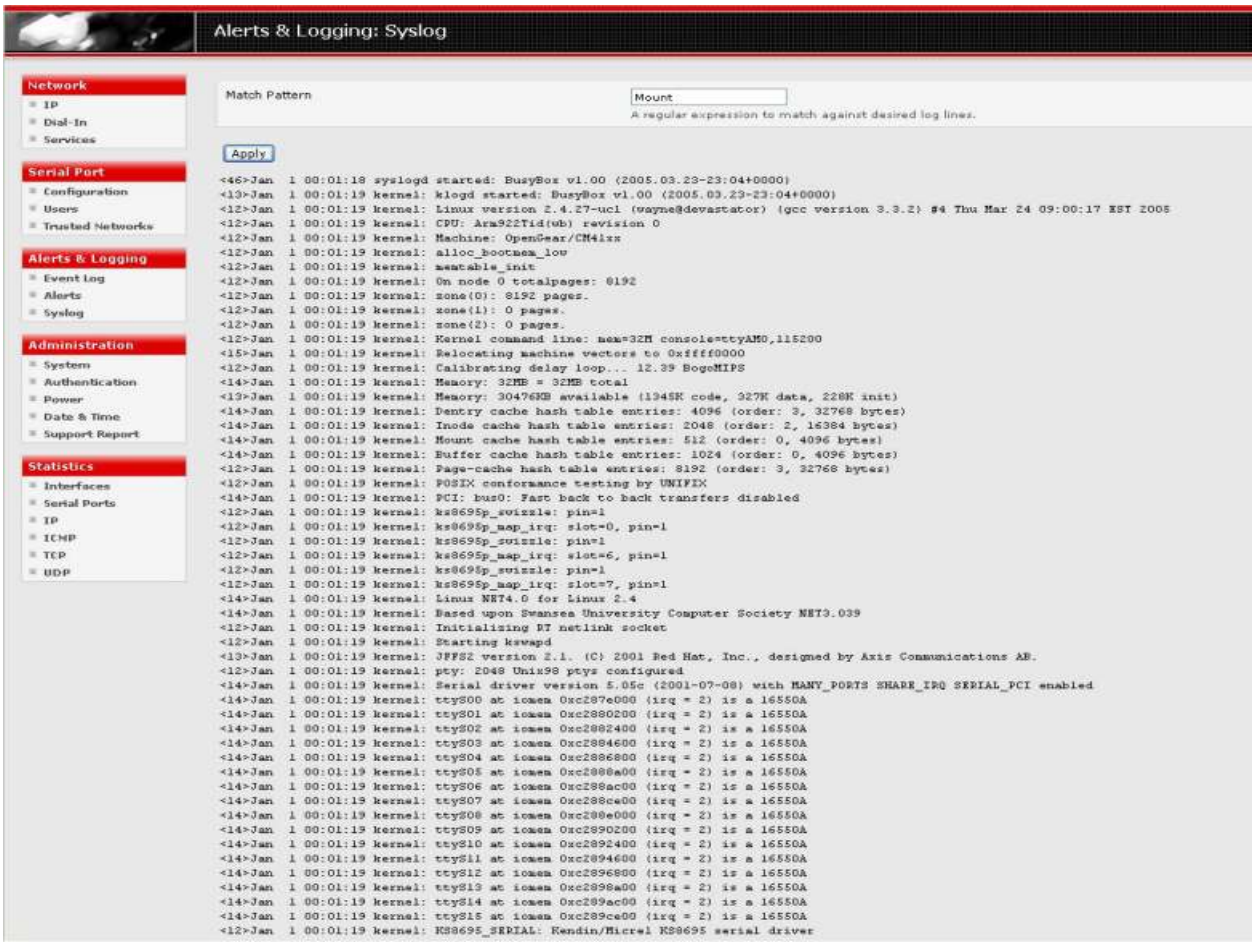
The Linux System Logger maintains a record of all system messages and errors.

- Select **Alerts & Logging: Syslog**



The syslog record can be redirected to a remote Syslog Server:

- Enter the remote Syslog Server address and port details and click **Apply**


Alerts & Logging: Syslog
Network
IP
Dial-In
Services
Serial Port
Configuration
Users
Trusted Networks
Alerts & Logging
Event Log
Alerts
Syslog
Administration
System
Authentication
Power
Date & Time
Support Report
Statistics
Interfaces
Serial Ports
IP
ICMP
TCP
UDP
Match Pattern
Mount
A regular expression to match against desired log lines.
Apply
<46>Jan 1 00:01:18 syslogd started: BusyBox v1.00 (2005.03.23-23:04+0000)
<13>Jan 1 00:01:19 kernel: klogd started: BusyBox v1.00 (2005.03.23-23:04+0000)
<12>Jan 1 00:01:19 kernel: Linux version 2.4.27-ucl (wayne@devastator) (gcc version 3.3.2) #4 Thu Mar 24 09:00:17 EST 2005
<12>Jan 1 00:01:19 kernel: CPU: Arm922Tid(wb) revision 0
<12>Jan 1 00:01:19 kernel: Machine: OpenGear/CM41xx
<12>Jan 1 00:01:19 kernel: alloc_bootmem_low
<12>Jan 1 00:01:19 kernel: memtable_init
<12>Jan 1 00:01:19 kernel: On node 0 totalpages: 8192
<12>Jan 1 00:01:19 kernel: zone(0): 8192 pages.
<12>Jan 1 00:01:19 kernel: zone(1): 0 pages.
<12>Jan 1 00:01:19 kernel: zone(2): 0 pages.
<12>Jan 1 00:01:19 kernel: Kernel command line: mem=32M console=ttyAM0,115200
<15>Jan 1 00:01:19 kernel: Relocating machine vectors to 0xffff0000
<12>Jan 1 00:01:19 kernel: Calibrating delay loop... 12.39 BogoMIPS
<14>Jan 1 00:01:19 kernel: Memory: 32MB = 32MB total
<13>Jan 1 00:01:19 kernel: Memory: 30476KB available (1345K code, 327K data, 228K init)
<14>Jan 1 00:01:19 kernel: Dentry cache hash table entries: 4096 (order: 3, 32768 bytes)
<14>Jan 1 00:01:19 kernel: Inode cache hash table entries: 2048 (order: 2, 16384 bytes)
<14>Jan 1 00:01:19 kernel: Mount cache hash table entries: 512 (order: 0, 4096 bytes)
<14>Jan 1 00:01:19 kernel: Buffer cache hash table entries: 1024 (order: 0, 4096 bytes)
<12>Jan 1 00:01:19 kernel: Page-cache hash table entries: 8192 (order: 3, 32768 bytes)
<12>Jan 1 00:01:19 kernel: POSIX conformance testing by UNIFIX
<14>Jan 1 00:01:19 kernel: PCI: bus0: Fast back to back transfers disabled
<12>Jan 1 00:01:19 kernel: ks8695p_swizzle: pin=1
<12>Jan 1 00:01:19 kernel: ks8695p_map_irq: slot=0, pin=1
<12>Jan 1 00:01:19 kernel: ks8695p_swizzle: pin=1
<12>Jan 1 00:01:19 kernel: ks8695p_map_irq: slot=6, pin=1
<12>Jan 1 00:01:19 kernel: ks8695p_swizzle: pin=1
<12>Jan 1 00:01:19 kernel: ks8695p_map_irq: slot=7, pin=1
<14>Jan 1 00:01:19 kernel: Linux NET4.0 for Linux 2.4
<14>Jan 1 00:01:19 kernel: Based upon Swansea University Computer Society NET3.039
<12>Jan 1 00:01:19 kernel: Initializing RT netlink socket
<12>Jan 1 00:01:19 kernel: Starting kswapd
<13>Jan 1 00:01:19 kernel: JFFS2 version 2.1. (C) 2001 Red Hat, Inc., designed by Axis Communications AB.
<12>Jan 1 00:01:19 kernel: pty: 2048 Unix98 ptys configured
<14>Jan 1 00:01:19 kernel: Serial driver version 5.05c (2001-07-08) with MANY_PORTS SHARE_IRQ SERIAL_PCI enabled
<14>Jan 1 00:01:19 kernel: ttyS00 at iomem 0xc207e000 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS01 at iomem 0xc2880200 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS02 at iomem 0xc2882400 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS03 at iomem 0xc2884600 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS04 at iomem 0xc2886800 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS05 at iomem 0xc2888a00 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS06 at iomem 0xc288ac00 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS07 at iomem 0xc288ce00 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS08 at iomem 0xc288e000 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS09 at iomem 0xc2890200 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS10 at iomem 0xc2892400 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS11 at iomem 0xc2894600 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS12 at iomem 0xc2896800 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS13 at iomem 0xc2898a00 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS14 at iomem 0xc289ac00 (irq = 2) is a 16550A
<14>Jan 1 00:01:19 kernel: ttyS15 at iomem 0xc289ce00 (irq = 2) is a 16550A
<12>Jan 1 00:01:19 kernel: KS8695_SERIAL: Kendin/Micrel KS8695 serial driver

To view the local Syslog file:

- Select **Alerts & Logging: Syslog**

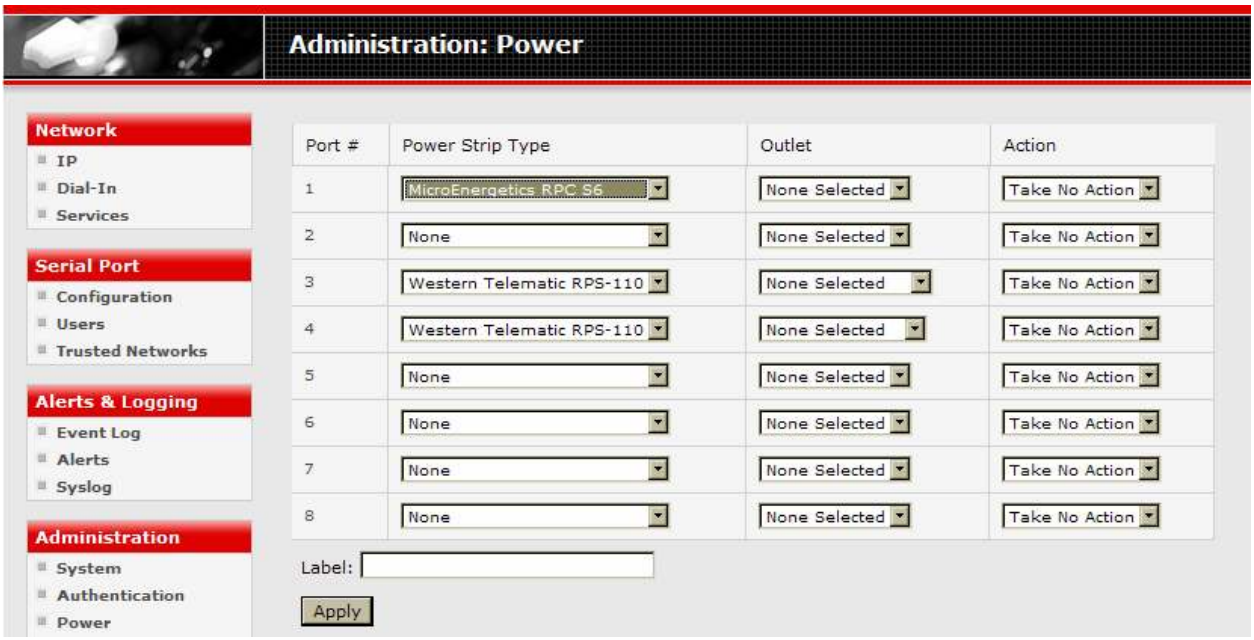
To make it easier to find information in the local Syslog file, a pattern matching filter tool is provided.

- Specify the **Match Pattern** that is to be searched for (*e.g.* the search for *Mount* is shown below) and click **Apply.** The Syslog will then be represented with only those entries that actually include the specified pattern

Chapter 8 Power Control

Introduction

The CM4000 console server enables Administrators to control attached power strips. This power control allows Administrators to remotely power on, power off, and power cycle the appliances, and services, that are connected to the power strips. Where possible the current status of the power strip appliance can also be read.

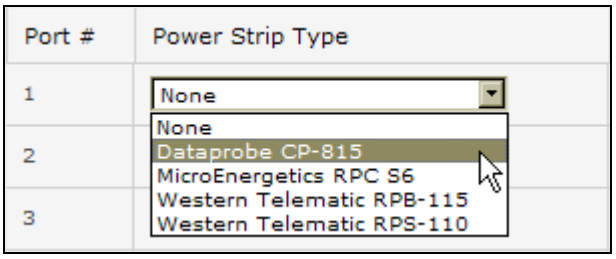


Configuring Power Strips

To set up power strip control, you must connect and configure the selected console server port to communicate with the power strip, then configure the power strip.

- Select the **Serial Port: Configuration** menu option and configure the selected console server port that will be connected to the power strip with the RS232 properties required by the power strip
- Serially connect the power strip to the port
- Go to the **Administration: Power** menu option

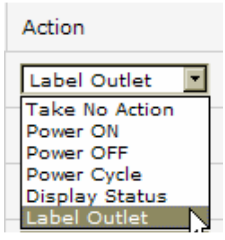
- Select the appropriate Power Strip for the connected Port
- Click **Apply**



Note The Management Console has support for a limited set of popular power-control devices. If your device is not on the default list it is fairly simple to add support for more devices, and this is covered in Chapter 11 (Advanced Configurations).

To ease management, you also can now optionally apply a text label to each of the power outlets on the power strip you have installed.

- Select the appropriate Power Strip for the connected Port
- Click **Apply**



Actioning Power Control

- Select the **Port / Power Strip** and the particular **Outlet** to be controlled
- Then select the desired **Action** to be taken (**Power ON**, **Power OFF**, **Power Cycle** or **Display Status**). You are only presented with those operations supported by the selected **Power Strip Type**.

Chapter 9 Authentication

Introduction

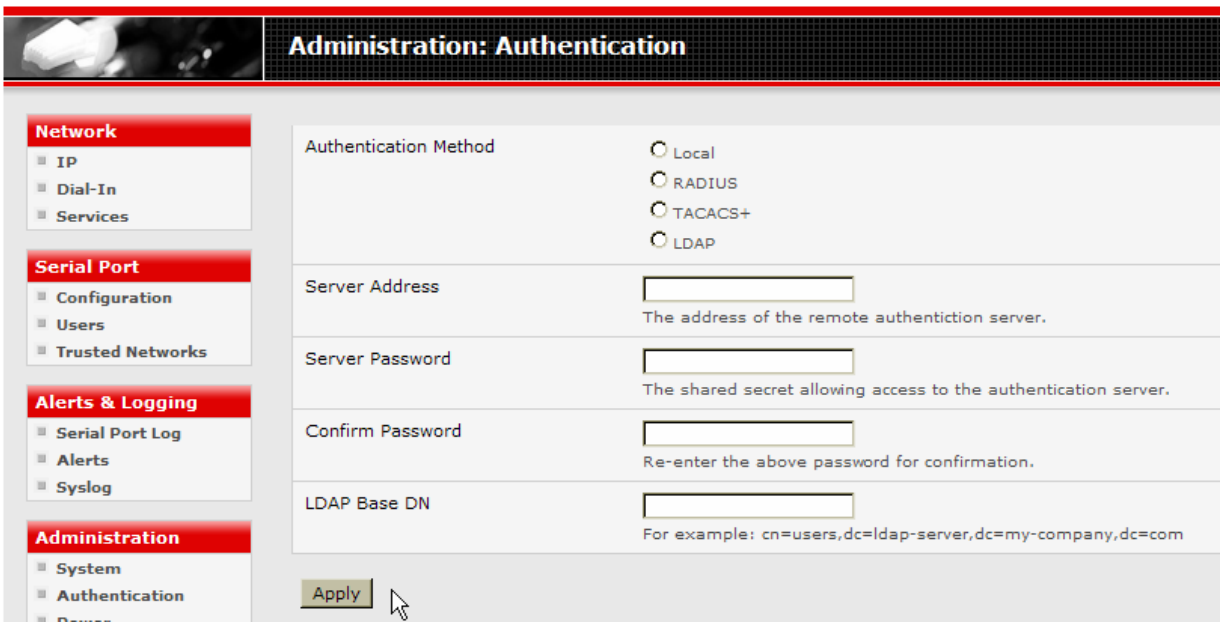
The CM4000 platform is a dedicated Linux computer, and it embodies a myriad of popular and proven Linux software modules for networking, secure access (OpenSSH) and communications (OpenSSL) and sophisticated user authentication (PAM, RADIUS, TACACS+ and LDAP).

- This chapter details how the administrator can use the Management Console to establish remote authentication for all User connections to ports on the CM4000
- This chapter also covers establishing a secure link to the Management Console using HTTPS and using OpenSSL and OpenSSH for establishing secure Administration connection to the CM4000

Remote Authentication Configuration

By default all access to the CM4000 uses local authentication (ie no remote authentication). The administration user (**root**) always uses local authentication, whether connected in band or out of band *via* the modem link.

The administrator can use the Management Console to set up remote authentication for all user connections to ports on the CM4000. The remote authentication database is then used to verify the username and password received from users. To enable remote authentication:



- Select **Administration: Authentication**
- Select if **Radius TACAS+** or **LDAP** authentication is to be used. For local authentication only, select **None**
- Enter the **Server Address** (IP or host name) of the remote server and the **Server Password**
- Click **Apply.** The selected remote authentication will now be used for all user access to console server ports

RADIUS The Remote Authentication Dial-In User Service (RADIUS) protocol was developed by Livingston Enterprises as an access server authentication and accounting protocol. The RADIUS server can support a variety of methods to authenticate a user. When it is provided with the username and original password given by the user, it can support PPP, PAP or CHAP, UNIX login, and other authentication mechanisms.

TACACS+ The Terminal Access Controller Access Control System (TACACS+) security protocol is a more recent protocol developed by Cisco. It provides detailed accounting information and flexible administrative control over the authentication and authorization processes. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide authentication, authorization, and accounting services independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon. There is a draft RFC detailing this protocol.

LDAP The Lightweight Directory Access Protocol (LDAP) is based on the X.500 standard, but significantly simpler and more readily adapted to meet custom needs. The core LDAP specifications are all defined in RFCs. LDAP is a protocol used to access information stored in an LDAP server.

PAM (Pluggable Authentication Modules)

The CM4000 supports RADIUS, TACACS+ and LDAP for two-factor authentication *via* PAM (Pluggable Authentication Modules). PAM is a flexible mechanism for authenticating users. Nowadays a number of new ways of authenticating users have become popular. The problem is that each time a new authentication scheme is developed; it requires all the necessary programs (login, ftpd *etc*.) to be rewritten to support it.

PAM provides a way to develop programs that are independent of authentication scheme. These programs need "authentication modules" to be attached to them at run-time in order to work. Which authentication module is to be attached is dependent upon the local system setup and is at the discretion of the local Administrator.

The CM4000 family supports PAM to which we have added the following modules for remote authentication:

RADIUS	- pam_radius_auth	(http://www.freeradius.org/pam_radius_auth/)
TACACS+	- pam_tacplus	(http://echelon.pl/pubs/pam_tacplus.html)
LDAP	- pam_ldap	(http://www.padl.com/OSS/pam_ldap.html)

Further modules can be added as required.

Note The above links point to the standard documentation. The implementation of PAM in CM4000 is the latest version of PAM from http://www.kernel.org/pub/linux/libs/pam

The only changes are:

1) The config files are now in */etc/config* e.g.
/etc/config/pam.conf instead of /etc/pam.conf
/etc/config/pam.d instead of /etc/pam.d

2) We have added extra modules for remote authentication:
RADIUS - pam_radius_auth (http://www.freeradius.org/pam_radius_auth/)
TACACS+ - pam_tacplus (http://echelon.pl/pubs/pam_tacplus.html)
LDAP - pam_ldap (http://www.padl.com/OSS/pam_ldap.html)

3) Limited set of modules supported:

pam_debug.so	pam_ldap.so	pam_radius_auth.so	pam_time.so
pam_deny.so	pam_limits.so	pam_rootok.so	pam_unix.so
pam_env.so	pam_localuser.so	pam_shells.so	pam_warn.so
pam_filter.so	pam_motd.so	pam_succeed_if.so	
pam_issue.so	pam_permit.so	pam_tacplus.so	

For further information on configuring remote RADIUS, TACACS+ or LDAP servers can be found at the following sites:

RADIUS

http://www.microsoft.com/technet/prodtechnol/windowsserver2003/library/DepKit/d4fe8248-eecd-49e4-88f6-9e304f97fefc.mspx

http://www.cisco.com/en/US/tech/tk59/technologies_tech_note09186a00800945cc.shtml

http://www.freeradius.org/

TACACS+

http://www.cisco.com/en/US/tech/tk59/technologies_tech_note09186a0080094e99.shtml

http://www.cisco.com/en/US/products/sw/secursw/ps4911/products_user_guide_chapter09186a00800eb6d6.html

http://cio.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113ed_cr/secur_c/scprt2/sctplus.htm

LDAP

http://www.ldapman.org/articles/intro_to_ldap.html

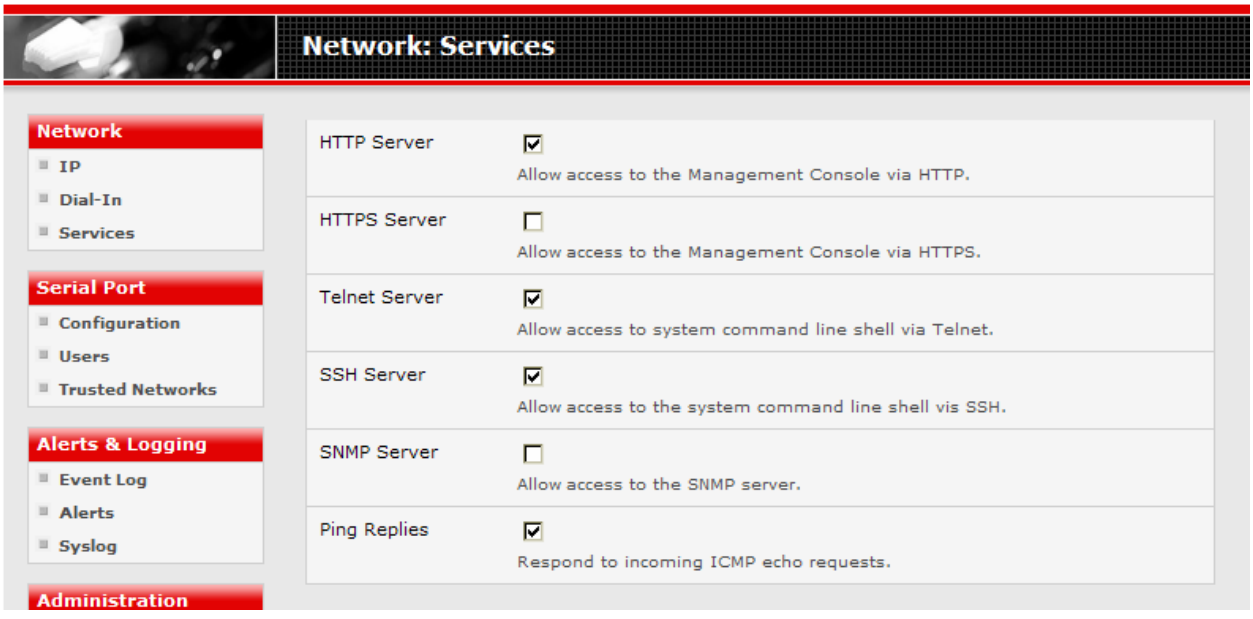
http://www.ldapman.org/servers.html

http://www.linuxplanet.com/linuxplanet/tutorials/5050/1/

http://www.linuxplanet.com/linuxplanet/tutorials/5074/4/

Secure Management Console Access

If you selected **HTTPS Server** in **Network: Services** then this will enable you, the Administrator, to establish a secure browser connection to the CM4000 Management Console. To securely access the Management Console from a network connected PC or workstation, you must:



- Activate your preferred browser and enter **https:// CM4000's IP address** For example, if the CM4000 has been set up with an IP address of 200.122.0.12 you need to type https:// 200.122.0.12 in your address bar

- Your browser may respond with a message that verifies the security certificate is valid but notes that it is not necessarily verified by a certifying authority. To proceed you need to click ‘yes’ if you are using Internet Explorer or select ‘accept this certificate permanently’ or ‘temporarily’ if you are using Mozilla Firefox.

- You will then be prompted for the administrator account and password as normal.

When you have a secure HTTPS connection in place the “SSL secured” icon will appear at the bottom of the browser screen. You can verify the level of encryption in place by clicking on this icon.

When you first enable and connect *via* HTTPS it is normal that you may receive a certificate warning. The default SSL certificate in your CM4000 is embedded during testing and it is not signed by a recognized third party certificate authority (rather it is signed by our own signing authority). These warnings do not affect the encryption protection you have against eavesdroppers.

Note More detailed information on issuing certificates and configuring HTTPS can be found in Chapter 11 - Advanced

Chapter 10 System Management

Introduction

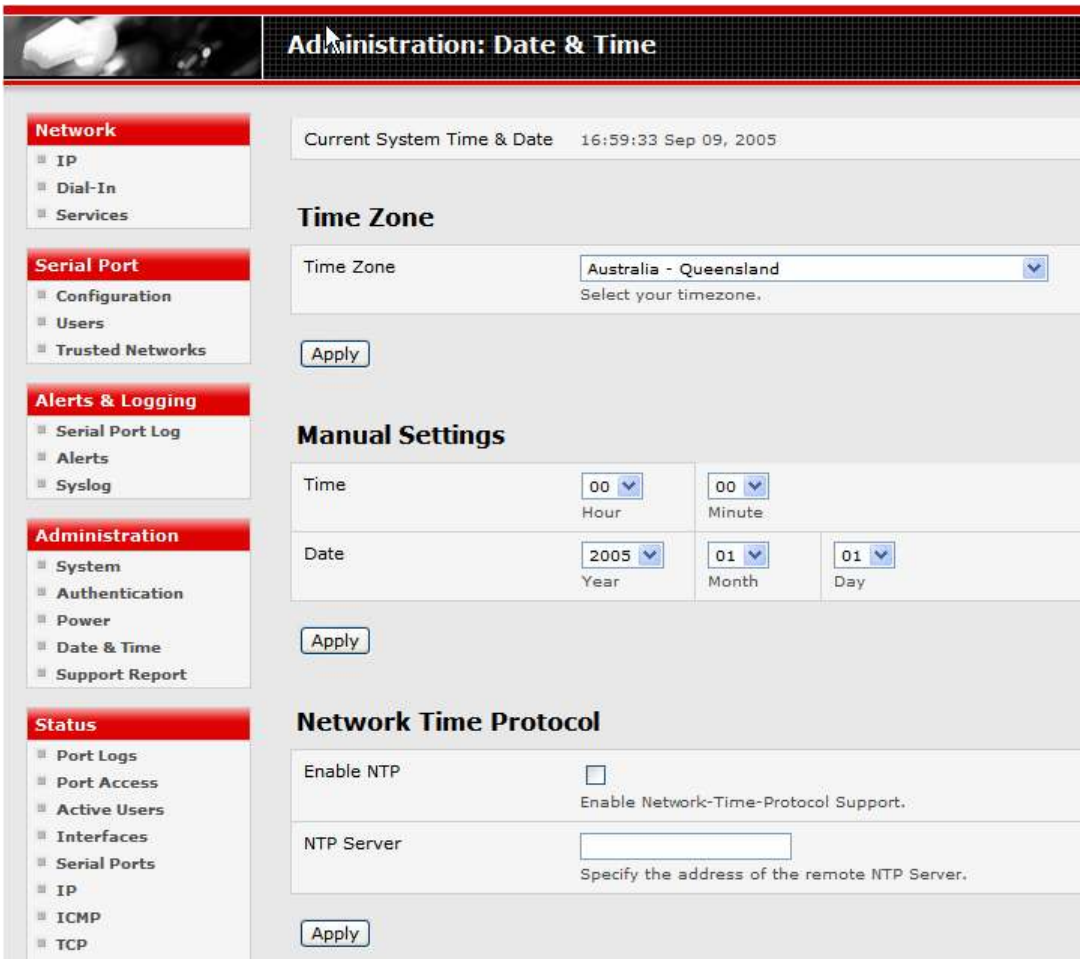
This chapter describes how to perform a range of general system management tasks:

- Configuring the Date, Time and NTP
- Applying *Soft* and *Hard* Resets to the console server
- Reflashing the Firmware
- Viewing Support Reports
- Monitoring Statistics

Configure Date and Time

It is recommended that you set the local Date and Time in the CM4000 as soon as it is configured. Some features, like Syslog and NFS logging, use the system time for time-stamping log entries, while certificate generation depends on a correct *Timestamp* to check the validity period of the certificate.

- Select the **Administration: Date & Time** menu option
- Manually set the **Year**, **Month**, **Day**, **Hour** and **Minute** using the **Date** and **Time** selection boxes, then click **Apply**



Configure NTP

The console server can synchronize its system time with a remote time server using the Network Time Protocol (NTP). Configuring the NTP time server ensures that the CM4000 clock will be accurate soon after the Internet connection is established. Also if NTP is not used, the system clock will be reset randomly every time the CM4000 is powered up. To set the system time using NTP:

- Select the **Enable NTP** checkbox on the **Network Time Protocol** page
- Enter the IP address of the remote **NTP Server** and click **Apply**

You must now also specify your local time zone so the system clock can show local time (and not UTP):

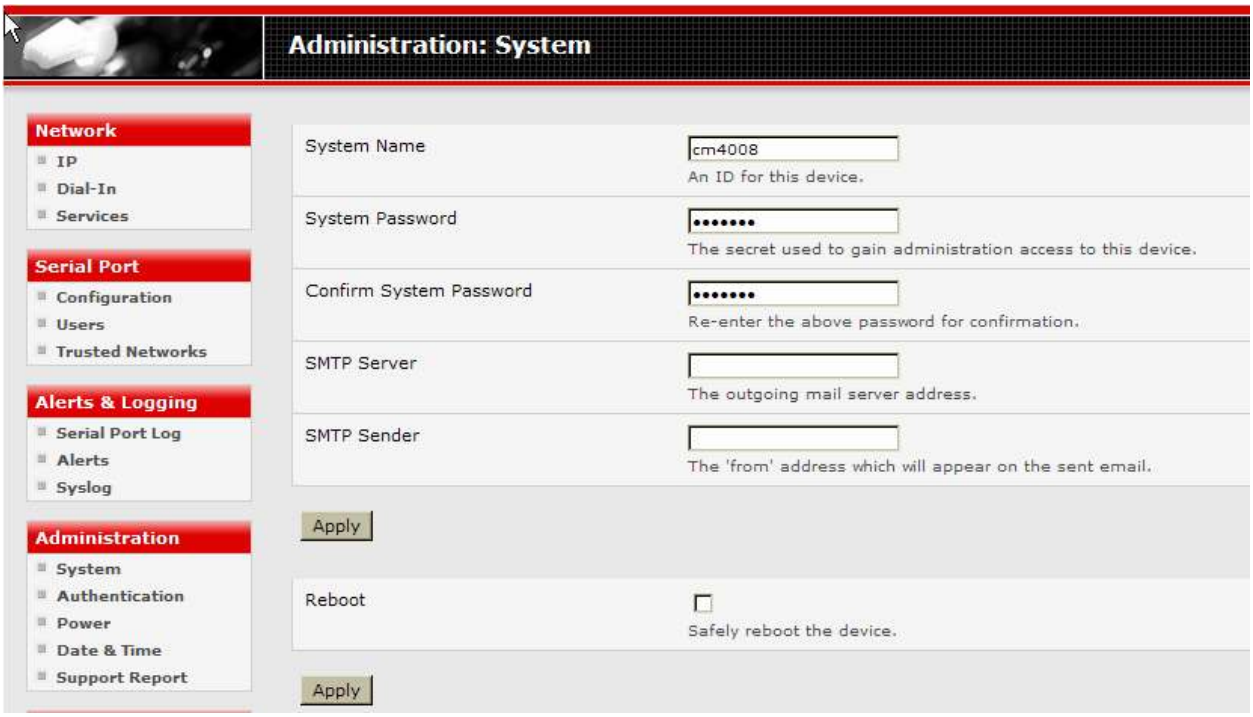
- Set your appropriate region/locality in the **Time Zone** selection box
- Click **Apply**

Soft and Hard Reset

A *soft* reset is affected by:

- Selecting **Reboot** in the **Administration: System** menu and clicking **Apply**

The CM4000 reboots with all user defined settings (*e.g.* the assigned network IP address) preserved. However this *soft* reset does disconnect all users and end any SSH sessions that had been established.



A *soft* reset will also be affected when you switch OFF power from the CM4000, and then switch the power back ON. However if you cycle the power and the unit is writing to flash you could corrupt or lose data, so the software reboot is the safer option.

Note The Management Console uses the Busybox reboot command to shutdown the system securely /safely and bring it back up again. You can execute this command directly from the CM4000 Linux command line by entering ***reboot [-d]*** where *-d* is the delay interval for rebooting.

A *hard* erase (*hard reset*) will reset the CM4000 back to its factory default settings. The simplest method to perform a *hard* erase, and clear all the CM4000 appliance's stored configuration information is by pushing the *Erase* button on the rear panel **twice**.

A ball point pen or bent paper clip is a suitable tool for performing this procedure. Do not use a graphite pencil. Depress the button gently **twice** (within a 5 second period) while the unit is powered ON.

This *hard* erase will clear all custom settings and return the unit back to factory default settings (*i.e.* the IP address will be reset to 192.168.0.1).

You will be prompted to log in and must enter the default administration user name and administration password:

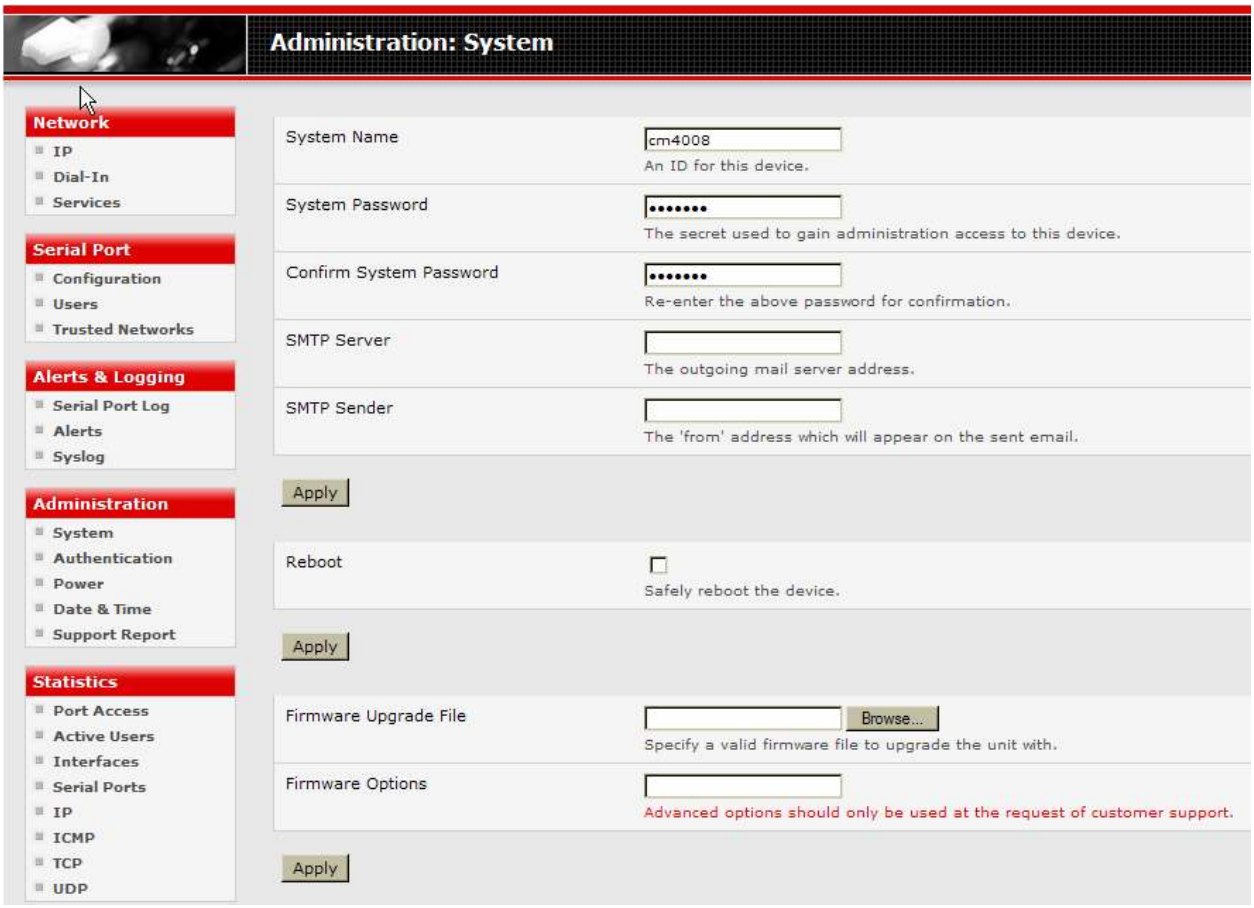
Username: **root**
Password: **default**

Upgrading the CM4000 Firmware

Before upgrading you should ascertain if you are already running the most current firmware in your console server. Your CM4000 will not allow you to upgrade to the same or an earlier version.

- Select **Administration: Support Report** and note the **Firmware Version**

- To upgrade, you first must download the latest firmware image from ftp://ftp.opengear.com For CM4008 download the *cm4008.flash* file and for both CM4116 and CM4148 download *cm41xx.flash S*ave this downloaded file on to a system on the same subnet as the CM4000. Also download and read the *release_notes.txt* for the latest information



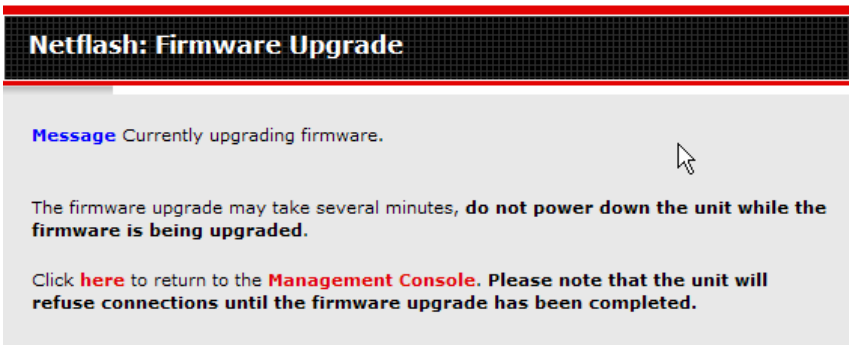
Then up load the firmware image file to your CM4000:

- Select **Administration: System**

- Specify the address and name of the downloaded Firmware Upgrade File, or **Browse** the local subnet and locate the downloaded file

> **Note** Any entry in the **Firmware Options** will change the operation of the firmware upload program, so, unless you are specifically advised otherwise by an Opengear Support Engineer, you must leave this field blank.

- Click **Apply** and the CM4000 appliance will undertake a soft reboot and commence upgrading the firmware. This process will take several minutes

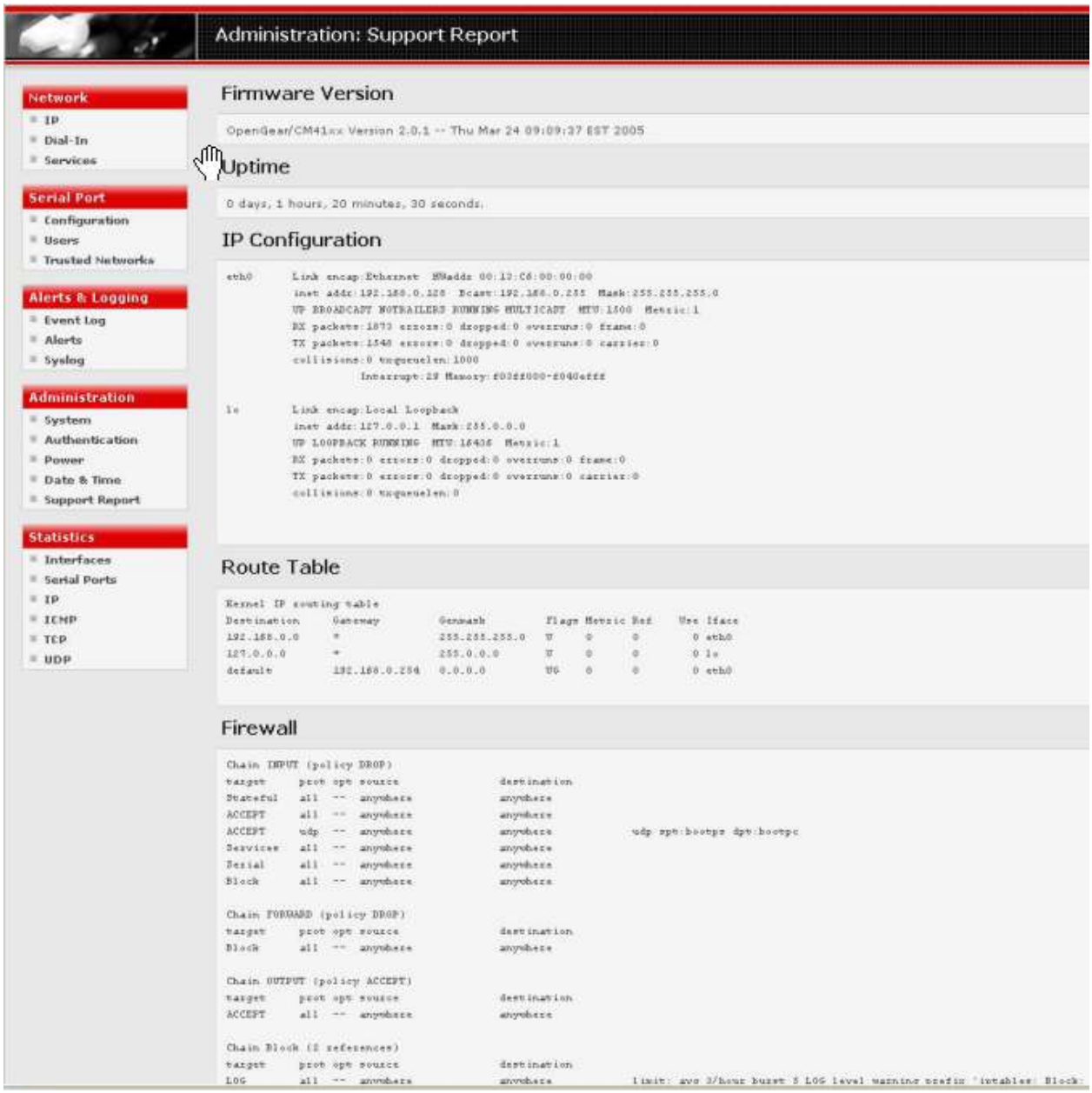

Netflash: Firmware Upgrade
Message Currently upgrading firmware.
The firmware upgrade may take several minutes, do not power down the unit while the firmware is being upgraded.
Click here to return to the Management Console. Please note that the unit will refuse connections until the firmware upgrade has been completed.

- After the firmware upgrade has completed, click **here** to return to the Management Console. Your CM4000 will have retained all its pre-upgrade configuration information

> ***WARNING***
>
> ***If the flash upgrade is interrupted (e.g. the power goes down) the CM4000 will stop functioning and will be unusable until its flash is factory reprogrammed. User care is advised.***

Support Reports

The Support Report provides useful status information that will assist the Opengear technical support team to solve any problems you may experience with your CM4000.

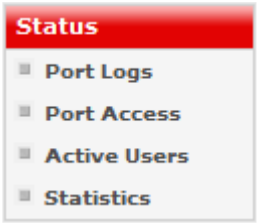


If you do experience a fault and have to contact the support team, ensure you include the Support Report with your email support request. The Support Report should be generated when the issue is occurring, and attached in plain text format.

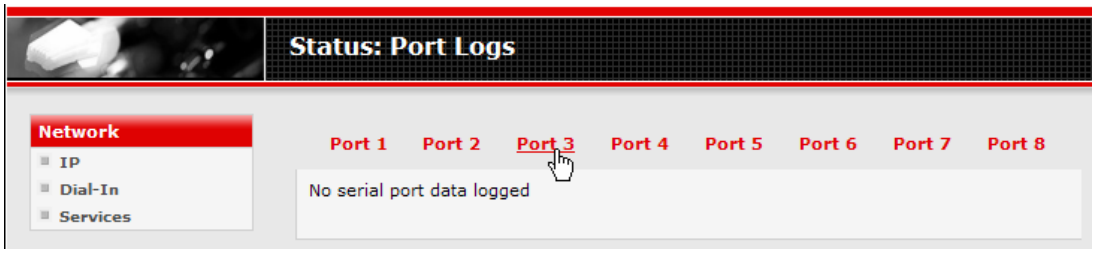
- Select the **Administration: Support Report** menu option and you will be presented with a snapshot of your console server's status
- Save the file as a text file and attach it to your support email

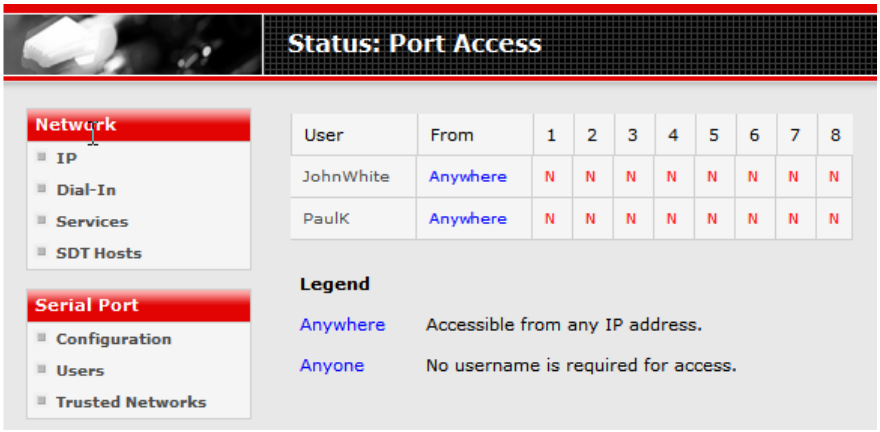
Status

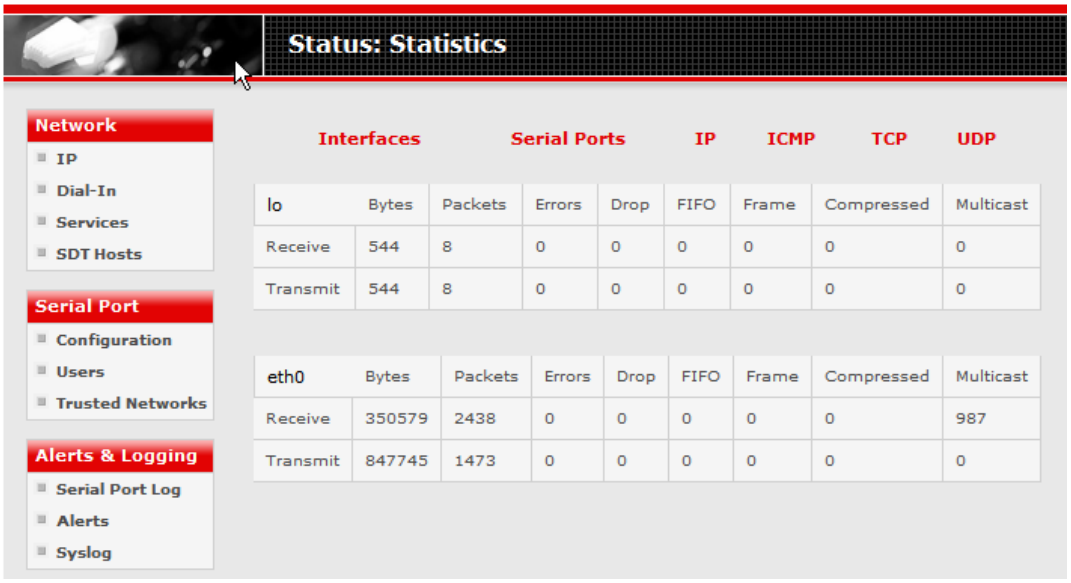
The Status reports provide a snapshot of the data traffic and other activities and operations of your console server. This information may be of assistance in diagnosing any problems you may experience *e.g.*



- the Port Logs keeps a record of all the local serial data traffic to and from each port
- the administrator can see which users have active sessions and who can access which ports, with the Port Access and Active Users






Status: Statistics
Network
IP
Dial-In
Services
SDT Hosts
Serial Port
Configuration
Users
Trusted Networks
Alerts & Logging
Serial Port Log
Alerts
Syslog
Interfaces
Serial Ports
IP
ICMP
TCP
UDP
lo Bytes Packets Errors Drop FIFO Frame Compressed Multicast
Receive 544 8 0 0 0 0 0 0
Transmit 544 8 0 0 0 0 0 0
eth0 Bytes Packets Errors Drop FIFO Frame Compressed Multicast
Receive 350579 2438 0 0 0 0 0 987
Transmit 847745 1473 0 0 0 0 0 0

Chapter 11 Basic Configuration - Linux Commands

Introduction

For those who prefer to configure their CM4000 at the Linux command line level (rather than use a browser and the Management Console), this chapter describes getting command line access and using the ***config*** tool to manage the system and configure the ports *etc.* from the command line:

- Administration Configuration (System Settings and Authentication Configuration)
- Date and Time Configuration (Manually Change Clock Settings and Network Time Protocol Time Zone)
- Network Configuration (Static and DHCP IP Configuration, Dial-in Configuration and Services Configuration)
- Serial Port Configuration (Serial Port Settings, Supported Protocol Configuration, Users and Trusted Networks)
- Event Logging Configuration (Remote Serial Port Log Storage and Alert Configuration)

The *config* documentation in this chapter walks thru basic configuration (in line with what can be done with the Management Console). For advanced and custom configurations using other standard commands, refer to Chapter 12.

The CM4000 runs a standard Linux kernel so it is also possible to configure the console server using other standard Linux and Busybox commands and applications (*ifconfig, gettyd, stty etc*.) However doing this will not guarantee these changes are permanent.

WARNING

This chapter is not intended to teach you Linux. We assume you already have a certain level of understanding before you execute Linux kernel level commands.

The Linux Command line

- Power up the CM4000 and connect the “terminal” device:
  - If you are connecting using the serial line, plug a serial cable between the CM4000 local DB-9 port and terminal device. Configure the serial connection of the “terminal” device/program you are using to 115200bps, 8 data bits, no parity and one stop bit. If you are using a program running on a Windows PC as the terminal device, then the cable is made up from a Cat5 UTP (#440016) cable and two DB-9 to RJ-45 adapters (#319000 and #319001)
  - If you are connecting over the LAN then you will need to interconnect the Ethernet ports and direct your terminal emulator program to the IP address of the CM4000 (192.168.0.1 by default)
- Log on to the CM4000 by pressing ‘return’ a few times. The CM4000 will request a user name and password. Enter the user name *root* and the password *default*. You should now see the command line prompt which is a hash (#)

The *config* Tool

Syntax

config [-ahv] [-d id] [-g id] [-p path] [-r configurator] [-s id=value]

Description

The config tool allows manipulation and querying of the system configuration from the command line. Using config, the new configuration can be activated by running the relevant configurator which performs the action necessary to make the configuration changes live.

Configuration elements which can be changed are specified by a unique '.' separated name. For example the configuration file version is identified as 'config.version'.

The config tool is designed to perform multiple actions from one command if need be, so if necessary options can be chained together.

Options

-a –run-all Run all registered configurators. This will perform every configuration synchronization action pushing all changes to the live system.

-h –help Display a brief usage message.

-v –verbose Log extra debug information

-d –del=id Remove the given configuration element specified by a '.' separated identifier.

-g –get=id Display the value of a configuration element.

-p –path=file Specify an alternate configuration file to use. The default file is located at */etc/config/config.xml*

-r –run=configurator Run the specified registered configurator. Registered configurators are alerts, auth, dialin, eventlog, ipconfig, power, serialconfig, services, systemsettings, time and users.

-s --set=id=value Change the value of configuration element specified by a '.' separated identifier.

Administration Configuration

System Settings

To change system settings to the following values:

System Name	og.mydomain.com
System Password (root account)	secret
System SMTP Server	192.168.0.124
System SMTP Sender	og@mydomain.com

The following commands must be issued:

```
# /bin/config --set=config.system.name=og.mydomain.com
# /bin/config --set=config.system.password=secret
# /bin/config --set=config.system.smtp.server=192.168.0.124
# /bin/config --set=config.system.smtp.sender=og@mydomain.com
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=systemsettings
```

Authentication Configuration

You can configure the system remote authentication with the following settings:

Remote Authentication Method	LDAP
Server IP Address	192.168.0.32
Server Password	Secret
LDAP Base Node	Some base node

By issuing the following commands:

```
# /bin/config --set=config.auth.type=LDAP
# /bin/config --set=config.auth.server=192.168.0.32
# /bin/config --set=config.auth.password=Secret
# /bin/config --set="config.auth.ldap.basenode=some base node"
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=auth
```

Date and Time Configuration

Manually Change Clock Settings

To change the running system time you need to issue the following commands:

```
#  date 092216452005.05        Format is MMDDhhmm[[CC]YY][.ss]
```

Then the following command will save this new system time to the hardware clock:

```
# /bin/hwclock -systohc
```

Alternately to change the hardware clock time you need to issue the following commands:

```
# /bin/hwclock --set --date=092216452005.05
```

```
Where the format is MMDDhhmm[[CC]YY][.ss]
```

Then the following command will save this new hardware clock time as the system time:

```
# /bin/hwclock –hctosys
```

Network Time Protocol

To enable NTP using a server at pool.ntp.org issue the following commands:

```
# /bin/config --set=config.ntp.enabled=on
# /bin/config --set=config.ntp.server=pool.ntp.org
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=time
```

Time Zone

To change the system time zone USA eastern standard time you need to issue the following commands:

```
# /bin/config --set=config.system.timezone=US/Eastern
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=time
```

Network Configuration

IP Configuration

DHCP

To enable a DHCP client on the LAN interface (eth0) from the console server command line:

```
# /bin/config --set=config.interfaces.eth0.mode=dhcp
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=ipconfig
```

Note: “/bin/config” commands can be combined into one command for convenience.

Please note that supported interface modes are 'dhcp' and 'static'.

Static

To set static configuration on the LAN interface with the following attributes:

IP Address:	192.168.1.100
Network Mask:	255.255.255.0
Default Gateway:	192.168.1.1
Primary DNS:	192.168.1.254
Secondary DNS:	10.1.0.254

You would need to issue the following commands from the command line:

```
# /bin/config --set=config.interfaces.eth0.mode=static
# /bin/config --set=config.interfaces.eth0.address=192.168.1.100
# /bin/config --set=config.interfaces.eth0.netmask=255.255.255.0
# /bin/config --set=config.interfaces.eth0.gateway=192.168.1.1
# /bin/config --set=config.interfaces.eth0.dns1=192.168.1.254
# /bin/config --set=config.interfaces.eth0.dns2=10.1.0.254
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=ipconfig
```

Dial-in Configuration

To enable dial-in access on the DB9 serial port from the command line with the following attributes:

Local IP Address	172.24.1.1
Remote IP Address	172.24.1.2
Authentication Type:	MSCHAPv2
Serial Port Baud Rate:	115200
Serial Port Flow Control: Custom Modem Initialization:	Hardware ATQ0V1H0

You would need to issue the following commands from the command line to set system configuration:

```
# /bin/config --set=config.console.ppp.localip=172.24.1.1
# /bin/config --set=config.console.ppp.remoteip=172.24.1.2
# /bin/config --set=config.console.ppp.auth=MSCHAPv2
# /bin/config --set=config.console.ppp.enabled=on
# /bin/config --set=config.console.speed=115200
# /bin/config --set=config.console.flow=Hardware
# /bin/config --set=config.console.initstring=ATQ0V1H0
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=dialin
```

Please note that supported authentication types are 'None', 'PAP', 'CHAP' and 'MSCHAPv2'.

Supported serial port baud-rates are '9600', '19200', '38400', '57600', '115200', and '230400'.

Supported parity values are 'None', 'Odd', 'Even', 'Mark' and 'Space'.

Supported data-bits values are '8', '7', '6' and '5'.

Supported stop-bits values are '1', '1.5' and '2'.

Supported flow-control values are 'Hardware', 'Software' and 'None'.

If you do not wish to use out-of-band dial-in access please note that the procedure for enabling start-up messages on the console port is covered in Chapter 12 - Accessing the Console Port.

Services Configuration

You can manually enable or disable network servers from the command line. For example if you wanted to guarantee the following server configuration:

HTTP Server	Enabled
HTTPS Server	Disabled
Telnet Server	Disabled

SSH Server	Enabled
SNMP Server	Disabled
Ping Replies (Respond to ICMP echo requests)	Disabled

You would need to issue the following commands from the command line to set system configuration:

```
# /bin/config --set=config.services.http.enabled=on
# /bin/config --del=config.services.https.enabled
# /bin/config --del=config.services.telnet.enabled
# /bin/config --set=config.services.ssh.enabled=on
# /bin/config --del=config.services.snmp.enabled
# /bin/config --del=config.services.pingreply.enabled
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=services
```

Note: “/bin/config” commands can be combined into one command for convenience.

Serial Port Configuration

Serial Port Settings

To setup serial port 5 to use the following properties:

Baud Rate	115200
Parity	None
Data Bits	8
Stop Bits	1
Flow Control	Software

To enable a DHCP client on the LAN interface (eth0) from the console server command line:

```
# /bin/config --set=config.ports.port5.speed=115200
# /bin/config --set=config.ports.port5.parity=None
# /bin/config --set=config.ports.port5.charsize=8
```

```
# /bin/config --set=config.ports.port5.stop=1
# /bin/config --set=config.ports.port5.flow=Software
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=serialconfig
```

Note that supported serial port baud-rates are '9600', '19200', '38400', '57600', '115200', and '230400'.

Supported parity values are 'None', 'Odd', 'Even', 'Mark' and 'Space'.

Supported data-bits values are '8', '7', '6' and '5'.

Supported stop-bits values are '1', '1.5' and '2'.

Supported flow-control values are 'Hardware', 'Software' and 'None'.

Supported Protocol Configuration

To ensure remote access to serial port 5 is configured as follows:

Telnet Access LAN	Disabled
SSH Access LAN	Enabled
Raw TCP *via* LAN	Disabled

You would need to issue the following commands from the command line to set system configuration:

```
# /bin/config --set=config.ports.port5.ssh=on
# /bin/config --del=config.ports.port5.telnet
# /bin/config --del=config.ports.port5.tcp
The following command will synchronize the live system with the
new configuration.
# /bin/config --run=serialconfig
```

Note: “/bin/config” commands can be combined into one command for convenience.

Users

You can add a user to the system from the command line by following the following instructions:

Determine the total number of existing users (if you have no existing users) you can assume this is 0.

```
# /bin/config --get=config.users.total
```

This command should display:

```
config.users.total 1
```

Note that if you see:

```
config.users.total
```

This means you have 0 users configured.

So your new user will be the existing total plus 1 so if the previous command gave you 0 then you start with user number 1, if you already have 1 user your new user will be number 2 *etc.*

If you want a user named “user1” with a password of “secret” who will have access to serial port 5 from the network you need to issue the these commands (assuming you have a previous user in place):

```
# /bin/config --set=config.users.user2.username=user1
# /bin/config --set=config.users.user2.password=secret
# /bin/config --set=”config.users.user2.description=The Second User”
# /bin/config --set=config.users.user2.port5=on
# /bin/config --set=config.users.total=2
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=users
```

Trusted Networks

You can further restrict remote access to serial ports based on the source IP address. To configure this *via* the command line you need to do the following:

Determine the total number of existing trusted network rules (if you have no existing rules) you can assume this is 0.

```
# /bin/config --get=config.portaccess.total
```

This command should display:

```
config.portaccess.total 1
```

Note that if you see:

```
config.portaccess.total
```

This means you have 0 rules configured.

So your new rule will be the existing total plus 1 so if the previous command gave you 0 then you start with rule number 1, if you already have 1 rule your new rule will be number 2 *etc.*

If you want to restrict access to serial port 5 to computers from a single C class network 192.168.5.0, you need to issue the following commands (assuming you have a previous rule in place):

```
# /bin/config --set=config.portaccess.rule2.address=192.168.5.0
# /bin/config --
set=config.portaccess.rule2.netmask=255.255.255.0
# /bin/config --set="config.portaccess.rule2.description=foo
bar."
# /bin/config --set=config.portaccess.rule2.port5=on
# /bin/config --set=config.portaccess.total=2
```

Please note that this rule becomes live straight away.

Event Logging Configuration

a. Remote Serial Port Log Storage

To setup remote storage of serial port 5 log to a remote Windows share with the following properties:

IP Address	192.168.0.254
Directory	C:\\opengear\logs\
Username	cifs_user
Password	secret
Logging level	2 (input/output logging as well as user connections & disconnections)

The following commands must be issued:

```
# /bin/config --set=config.eventlog.server.type=cifs
# /bin/config --set=config.eventlog.server.address=192.168.0.254
# /bin/config --set=config.eventlog.server.path=/opengear/logs
# /bin/config --set=config.eventlog.server.username=cifs_user
# /bin/config --set=config.eventlog.server.password=secret
# /bin/config --set=config.ports.port5.loglevel=2
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=eventlog
```

Note that supported remote storage server types are 'None', 'cifs', 'nfs' and 'syslog'.

Supported port logging levels are '0', '1' and '2'.

b. Alert Configuration

You can add an email alert to the system from the command line by following these instructions:

Determine the total number of existing alerts (if you have no existing alerts) you can assume this is 0.

```
# /bin/config --get=config.alerts.total
```

This command should display output similar to:

```
config.alerts.total 1
```

Note that if you see:

```
config.alerts.total
```

This means you have 0 alerts configured.

So your new alert will be the existing total plus 1 so if the previous command gave you 0 then you start with user number 1, if you already have 1 alert your new alert will be number 2 *etc.*

To configure an email alert to be sent to alert1@domain.org when the regular expression “Cpu.*0.0% id,” matches logging on serial port 5 you would need to issue the following commands (Assuming you have 1 previous alert in place):

```
# /bin/config --set=config.alerts.alert2.email=alert1@domain.com
# /bin/config --set=”config.alerts.alert2.pattern=.*0.0% id,”
# /bin/config --set=config.alerts.alert2.port5=on
```

```
# /bin/config --del=config.alerts.total=2
```

The following command will synchronize the live system with the new configuration.

```
# /bin/config --run=alerts
```

Chapter 12 Advanced Configuration

Introduction

This chapter documents the ***portmanager*** application, which was developed by Opengear for console server serial port management, and gives examples of its use:

- Portmanager documentation
- Scripts and alerts
- Raw data access to the ports and modems

This chapter also describes details how to perform advanced and custom management tasks using Linux commands and script:

- *iptables* modifications and updating IP Filtering rules
- Modifying SNMP with *net-snmpd*
- Using secure SSH communications
- SSL, configuring HTTPS and issuing certificates
- Adding new Power Strips and Power Strip control

WARNING

This chapter is not intended to teach you Linux. We assume you already have a certain level of understanding before you execute Linux kernel commands. If you are not comfortable with this, we recommend getting some Linux technical help.

Advanced Portmanager

pmshell

The *pmshell* command acts similar to the standard *tip* or *cu* commands, but all serial port access is directed *via* the portmanager.

Example:

To connect to port 8 *via* the portmanager:

 # pmshell -l port08

pmshell Commands:

Once connected, the pmshell command supports a subset of the '~' escape commands that tip/cu support.

Send Break:

Typing the character sequence '~b' will generate a BREAK on the serial port.

Quit pmshell:

Typing the character sequence '~.' will exit from pmshell.

Set RTS to 1 run the command:

 # pmshell --rts=1

Show all signals

 # pmshell --signals
DSR=1 DTR=1 CTS=1 RTS=1 DCD=0

Read a line of text from the serial port:

 # pmshell --getline

pmchat

The *pmchat* command acts similar to the standard *chat* command, but all serial port access is directed *via* the portmanager.

Example:

To run a chat script *via* the portmanager:

 # pmchat -v -f /etc/config/scripts/port08.chat < /dev/port08

For more information on using *chat* (and *pmchat*) you should consult the UNIX man pages:

http://techpubs.sgi.com/library/tpl/cgi-bin/getdoc.cgi?coll=linux&db=man&fname=/usr/share/catman/man8/chat.8.html

pmusers

The *pmusers* command is used to query the portmanager for active user sessions.

Example:

To detect which users are currently active on which serial ports:

 # pmusers

This command will output nothing if there are no active users currently connected to any ports, otherwise it will respond with a sorted list of usernames per active port:

Port 1:
user1
user2
Port 2:
user1
Port 8:
user2

The above output indicates that a user named "*user1*" is actively connected to ports 1 and 2, while "*user2*" is connected to both ports 1 and 8.

Portmanager Daemon

Command line options

There is normally no need to stop and restart the daemon. To restart the daemon normally, just run the command:

 # portmanager

Supported command line options are:

Force portmanager to run in the foreground:

--nodaemon

Set the level of debug logging:

--loglevel={debug,info,warn,error,alert}

Change which configuration file it uses:

-c /etc/config/portmanager.conf

Signals

Sending a SIGHUP signal to the portmanager will cause it to re-read it's configuration file.

External Scripts and Alerts

The portmanager has the ability to execute external scripts on certain events. These events are:

I. When a port is opened by the portmanager.

   When the portmanager opens a port, it attempts to execute */etc/config/scripts/portXX.init* (where XX is the number of the port, *e.g.* 08). The script is run with STDIN and STDOUT both connected to the serial port.

   If the script cannot be executed, then portmanager will execute */etc/config/scripts/portXX.chat via* the chat command on the serial port.

II. When an alert occurs on a port.

   When an alert occurs on a port, the portmanager will attempt to execute */etc/config/scripts/portXX.alert* (where XX is the port number, e.g. 08)

   The script is run with STDIN containing the data which triggered the alert, and STDOUT redirected to /dev/null, NOT to the serial port. If you wish to communicate with the port, use *pmshell* or *pmchat* from within the script.

   If the script cannot be executed, then the alert will be mailed to the address configured in the system administration section.

Raw Access to Serial Ports

You can *tip* and *stty* to completely bypass the portmanager and have raw access to the serial ports.

When you run *tip* on a portmanager controlled port, portmanager closes that port, and stops monitoring it until tip releases control of it.

With *stty*, the changes made to the port only "stick" until that port is closed and opened again, so it is doubtful that people will want to use *stty* for more than initial debugging of the serial connection.

If you want to use stty to configure the port, you can put *stty* commands in */etc/config/scripts/portXX.init*, which gets run whenever portmanager opens the port.

Otherwise, any setup you do with *stty* will get lost when the portmanager opens the port. (the reason that portmanager sets things back to its *config* rather than using whatever is on the port, is so the port is in a known good state, and will work, no matter what things are done to the serial port outside of portmanager).

Accessing the Console Port

The console dial-in is handled by *mgetty*, with automatic PPP login extensions. *mgetty* is a smart *getty* replacement, designed to be used with hayes compatible data and data/fax modems. *mgetty* knows about modem initialization, manual modem answering (so your modem doesn't answer if the machine isn't ready), UUCP locking (so you can use the same device for dial-in and dial-out). *mgetty* provides very extensive logging facilities. All standard *mgetty* options are supported.

- Modem initialization strings

  To override the standard modem initialization string either use the Management Console (see chapter 5) or the command line config tool (see *Dial-In Configuration* in Chapter 11).

- Enabling Boot Messages on the Console

  If you are not using a modem on the DB9 console port and instead wish to connect to it directly via a Null Modem cable you may want to enable verbose mode allowing you to see the standard linux start-up messages. This can be achieved with the following commands:

/bin/config --set=config.console.debug=on # /bin/config --run=console # reboot

If at some point in the future you chose to connect a modem for dial-in out-of-band access the procedure can be reversed with the following commands.

/bin/config --del=config.console.debug # /bin/config --run=console # reboot

IP- Filtering

Standard IP-Filter configuration:

The system uses the *iptables* utility to provide a stateful firewall of LAN traffic. By default rules are automatically inserted to allow access to enabled services, and serial port access *via* enabled protocols. The commands which add these rules are contained in configuration files.

/etc/config/ipfilter

This is an executable shell script which is run whenever the LAN interface is brought up and whenever modifications are made to the *iptables* configuration as a result of CGI actions or the *config* command line tool.

The basic steps performed are as follows:

a) The current *iptables* configuration is erased.

b) If a customized IP-Filter script exists it is executed and no other actions are performed.

c) Standard policies are inserted which will drop all traffic not explicitly allowed to and through the system.

d) Rules are added which explicitly allow network traffic to access enabled services *e.g.* HTTP, SNMP *etc.*

e) Rules are added which explicitly allow traffic network traffic access to serial ports over enabled protocols *e.g.* Telnet, SSH and raw TCP.

Customizing the IP-Filter:

/etc/config/filter-custom

If the standard system firewall configuration is not adequate for your needs it can be bypassed safely by creating a file at */etc/config/filter-custom* containing commands to build a specialized firewall. This firewall script will be run whenever the LAN interface is brought up (including initially) and will override any automated system firewall settings.

Below is a simple example of a custom script which creates a firewall using the *iptables* command. Only incoming connections from computers on a C-class network 192.168.10.0 will be accepted when this script is installed at */etc/config/filter-custom* (Note that when this script is called any preexisting chains and rules have been flushed from *iptables*):

```
#/bin/sh
# Set default policies to drop any incoming or routable traffic
# and blindly accept anything from the 192.168.10.0 network.
iptables --policy FORWARD DROP
iptables --policy INPUT DROP
iptables --policy OUTPUT ACCEPT

# Allow responses to outbound connections back in.
iptables --append INPUT \
     --match state --state ESTABLISHED,RELATED --jump ACCEPT

# Explicitly accept any connections from computers on
# 192.168.10.0/24
iptables --append INPUT --source 192.168.10.0/24 --jump ACCEPT
```

Good documentation about using the *iptables* command can be found at the linux *netfilter* website http://netfilter.org/documentation/index.html

Resources

There are many high-quality tutorials and HOWTOs available *via* the *netfilter* website, in particular peruse the tutorials listed on the *netfilter* HOWTO page. A list of useful web locations has been compiled for your convenience below:

Netfilter Homepage	http://netfilter.org
Netfilter/iptables Tutorials	http://netfilter.org/documentation/index.html#documentation-tutorials

Modifying SNMP Configuration

/etc/config/snmpd.conf

The *net-snmpd* is an extensible SNMP agent, which when enabled should run with a default configuration. Its behavior can be customized via the options in */etc/config/snmpd.conf.*

Changing standard system information such as system contact, name and location can be achieved by editing */etc/config/snmpd.conf* file and locating the following lines:

```
sysdescr        "opengear"
syscontact      root <root@localhost>(configure
                /etc/default/snmpd.conf)
sysname         Not defined (edit /etc/default/snmpd.conf)
syslocation     Not defined (edit /etc/default/snmpd.conf)
```

Simply change the values of *sysdescr, syscontact, sysname* and *syslocation* to the desired settings and restart *snmpd*.

The *snmpd.conf* provides is extremely powerful and too flexible to completely cover here. The configuration file itself is commented extensively and good documentation is available at the *net-snmp* website http://www.net-snmp.org, specifically:

Man Page:	http://www.net-snmp.org/docs/man/snmpd.conf.html
FAQ:	http://www.net-snmp.org/docs/FAQ.html
Net-SNMPD Tutorial:	http://www.net-snmp.org/tutorial/tutorial-5/demon/snmpd.html

Secure Shell (SSH) Support

Popular TCP/IP applications such as telnet, rlogin, ftp, and others transmit their passwords unencrypted. Doing this across the Internet can have catastrophic consequences. It leaves the door open for eavesdropping, connection hijacking, and other network-level attacks.

Secure Shell (SSH) is a program to log into another computer over a network, to execute commands in a remote machine, and to move files from one machine to another. It provides strong authentication and secure communications over insecure channels.

OpenSSH, the de facto open source SSH application, encrypts all traffic (including passwords) to effectively eliminate these risks. Additionally, OpenSSH provides a myriad of secure tunneling capabilities, as well as a variety of authentication methods.

OpenSSH is the port of OpenBSD's excellent OpenSSH[0] to Linux and other versions of Unix. OpenSSH is based on the last free version of Tatu Ylonen's sample implementation with all patent-encumbered algorithms removed (to external libraries), all known security bugs fixed, new features reintroduced and many other clean-ups. OpenSSH has been created by Aaron Campbell, Bob Beck, Markus Friedl, Niels Provos, Theo de Raadt, and Dug Song. It has a homepage at http://www.openssh.com/

The only changes in the CM4000 SSH implementation are:

- PAM support
- EGD[1]/PRNGD[2] support and replacements for OpenBSD library functions that are absent from other versions of UNIX
- The config files are now in */etc/config. e.g.*
    - */etc/config/sshd_config* instead of */etc/sshd_config*
    - */etc/config/ssh_config* instead of */etc/ssh_config*
    - */etc/config/users/<username>/.ssh/* instead of */home/<username>/.ssh/*

Configuring SSH Public Key Authentication

This section describes how to generate and configure SSH keys.

Generating Keys

The following commands can be issued on a Linux host to produce a DSA public/private key pair:

 # ssh-keygen -t dsa

T he command will prompt you for a path to store the keys
(it will default to ~/.ssh/id_dsa) and a passphrase.

This will produce two files, id_dsa.pub (the public key) and id_dsa
(the private key). Full documentation for the ssh-keygen command can be found at:

http://www.openbsd.org/cgi-bin/man.cgi?query=ssh-keygen

Installing Keys

If you have existing SSH keys, you can skip the above Generating Keys step and install them "as is".

The public key can be installed on the unit remotely from the linux host with
the scp utility as follows:

Assuming the user on the Management Console is called "fred"; the IP address of the CM4000 is 192.168.0.1 (default); and the public key is on the *linux/unix* computer in *~/.ssh/id_dsa.pub*. Execute the following command on the *linux/unix* computer:

scp ~/.ssh/id_dsa.pub \root@192.168.0.1:/etc/config/users/fred/.ssh/authorized_keys

The authorized_keys file on the CM4000 needs to be owned by "fred", so login to the Management Console as **root** and type:

chown fred /etc/config/users/fred/.ssh/authorized_keys

More documentation on OpenSSH can be found at:

http://openssh.org/portable.html
http://www.openbsd.org/cgi-bin/man.cgi?query=ssh&sektion=1
http://www.openbsd.org/cgi-bin/man.cgi?query=sshd

Secure Sockets Layer (SSL) Support

Secure Sockets Layer (SSL) is a protocol developed by Netscape for transmitting private documents *via* the Internet. SSL works by using a private key to encrypt data that's transferred over the SSL connection.

The CM4000 includes OpenSSL. The OpenSSL Project is a collaborative effort to develop a robust, commercial-grade, full-featured, and Open Source toolkit implementing the Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS v1) protocols as well as a full-strength general purpose cryptography library. The project is managed by a worldwide community of volunteers that use the Internet to communicate, plan, and develop the OpenSSL toolkit and its related documentation.

OpenSSL is based on the excellent SSLeay library developed by Eric A. Young and Tim J. Hudson. The OpenSSL toolkit is licensed under an Apache-style licence, which basically means that you are free to get and use it for commercial and non-commercial purposes subject to some simple license conditions. In the CM4000 OpenSSL is used primarily in conjunction with ‘http’ in order to have secure browser access to the GUI management console across insecure networks.

More documentation on OpenSSL is available from:

http://www.openssl.org/docs/apps/openssl.html

http://www.openssl.org/docs/HOWTO/certificates.txt

HTTPS

The Management Console can be served using HTTPS by running the webserver *via sslwrap*. The server can be launched on request using *inetd*.

The HTTP server provided is a slightly modified version of the *fnord-httpd* from http://www.fefe.de/fnord/

The SSL implementation is provided by the *sslwrap* application compiled with OpenSSL support. More detailed documentation can be found at http://www.rickk.com/sslwrap/

If your default network address is changed or the unit is to be accessed *via* a known Domain Name you can use the following steps to replace the default SSL Certificate and Private Key with ones tailored for your new address.

1. Generating an encryption key

To create a 1024 bit RSA key with a password issue the following command on the command line of a linux host with the *openssl* utility installed:

openssl genrsa -des3 -out ssl_key.pem 1024

2. Generating a self-signed certificate with OpenSSL

This example shows how to use OpenSSL to create a self-signed certificate. OpenSSL is available for most Linux distributions *via* the default package management mechanism. (Windows users can check http://www.openssl.org/related/binaries.html)

To create a 1024 bit RSA key and a self-signed certificate issue the following *openssl* command from the host you have *openssl* installed on:

*openssl req -x509 -nodes -days 1000 *
-newkey rsa:1024 -keyout ssl_key.pem -out ssl_cert.pem

You will be prompted to enter a lot of information. Most of it doesn't matter, but the "Common Name" should be the domain name of your computer (*e.g.* test.opengear.com). When you have entered everything, the certificate will be created in a file called *ssl_cert.pem*.

3. Installing the key and certificate

The recommended method for copying files securely to the CM4000 unit is with an SCP

(Secure Copying Protocol) client. The *scp* utility is distributed with OpenSSH for most Unices, while Windows users can use something like the PSCP command line utility available with PuTTY.

The files created in steps 1 and 2 can be installed remotely with the *scp* utility as follows:

scp ssl_key.pem root@<address of unit>:/etc/config/
scp ssl_cert.pem root@<address of unit>:/etc/config/

or using PSCP:

pscp -scp ssl_key.pem root@<address of unit>:/etc/config/
pscp -scp ssl_cert.pem root@<address of unit>:/etc/config/

PuTTY and the PSCP utility can be downloaded from
http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

More detailed documentation on the PSCP can be found:
http://the.earth.li/~sgtatham/putty/0.58/htmldoc/Chapter5.html#pscp

4. Launching the HTTPS Server

Note that the easiest way to enable the HTTPS server is from the web Management Console. Simply click the apropriate checkbox in **Network -> Services -> HTTPS Server** and the HTTPS server will be activated (assuming the *ssl_key.pem* & *ssl_cert.pem* files exist in the */etc/config* directory).

Alternatively *inetd* can be configured to launch the secure fnord server from the command line of the unit as follows.

Edit the *inetd* configuration file. From the unit command line:

vi /etc/config/inetd.conf

Append a line:

443 stream tcp nowait root sslwrap -cert /etc/config/ssl_cert.pem -key /etc/config/ssl_key.pem -exec /bin/httpd /home/httpd"

Save the file and signal *inetd* of the configuration change.

kill -HUP `cat /var/run/inetd.pid`

The HTTPS server should be accessible from a web client at a URL similar to

this: https://<common name of unit>

More detailed documentation about the *openssl* utility can be found at the website: http://www.openssl.org/

Power Strip Control

The CM4000 supports a limited set of power-control devices which can be configured using the Management Console as described in Chapter 8. However it is fairly simple to add support for more devices, or to customize the existing device support.

The **Administration: Power** page uses information contained in */etc/powerstrips.xml* to configure and control devices attached to a serial port. The configuration also looks for (and loads) */etc/config/powerstrips.xml* if it exists.

The user can add their own support for more devices by putting definitions for them into */etc/config/powerstrips.xml*. This file can be created on a host system and copied to the Management Console device using *scp*. Alternatively, login to the Management Console and use *ftp* or *wget* to transfer files.

Here is a brief description of the elements of the XML entries in */etc/config/powerstrips.xml*.

```
<powerstrip>
        <id>Name or ID of the device support</id>

        <outlet port="port-id-1">Display Port 1 in menu</outlet>
        <outlet port="port-id-2">Display Port 2 in menu</outlet>
        ...

        <on>script to turn power on</on>
        <off>script to power off</off>
        <cycle>script to cycle power</cycle>
        <status>script to write power status to /var/run/power-status</status>

        <speed>baud rate</speed>
        <charsize>character size</charsize>
        <stop>stop bits</stop>
        <parity>parity setting</parity>
</powerstrip>
```

The *id* appears on the web page in the list of available devices types to configure.

The outlets describe targets that the scripts can control. For example a power control board may control several different outlets. The port-id is the native name for identifying the outlet. This value will be passed to the scripts in the environment variable *outlet*, allowing the script to address the correct outlet.

There are four possible scripts: *on, off, cycle* and *status*

When a script is run, it's standard input and output is redirected to the appropriate serial port. The script receives the outlet and port in the *outlet* and *port* environment variables respectively.

The script can be anything that can be executed within the shell.

All of the existing scripts in */etc/powerstrips.xml* use the *pmchat* utility. *pmchat* works just like the standard unix "chat" program, only it ensures interoperation with the port manager.

The final options, *speed, charsize, stop* and *parity* define the recommended or default settings for the attached device.

Appendix A Linux Kernel and Source Code

The CM4000 platform is a dedicated Linux computer, optimized to provide secure access to serial consoles of critical server systems. Being based around uClinux (a small footprint but extensible Linux), it embodies a myriad, popular and proven Linux software modules for networking (NetFilter, IPTables), secure access (OpenSSH) and communications (OpenSSL) and sophisticated user authentication (PAM, RADIUS, TACACS+ and LDAP).

Many components of the CM4000 software are licensed under the GNU General Public License (version 2), which Opengear supports. You may obtain a copy of the GNU General Public License at http://www.fsf.org/copyleft/gpl.html. Opengear will provide source code for any of the components of the Software licensed under the GNU General Public License upon request.

Opengear CM4000 console servers are built on the 2.4 uClinux kernel as developed by the uClinux project. This is GPL code and source can be found: http://cvs.uclinux.org

Commands that have config files that can be altered:

portmanager	
inetd	
init	
ssh/sshd/scp/sshkeygen	
ucd-snmpd	http://www.ece.ucdavis.edu/ucd-snmp/
samba	
fnord (web server)	
sslwrap	

Commands you can run and do neat stuff with are:

loopback	
bash (shell)	
busybox	http://www.busybox.net/downloads/BusyBox.html (has lots of unix shell commands and tools)
chat	
dhcpcd	
ftp	
hd	
hwclock	
iproute	

iptables
netcat
ifconfig
mii-tool
netstat
route
openntpd
ping
portmap
pppd
routed
setserial
smtpclient
stty
stunel
tcpdump
tftp
tip
traceroute

A full list of the Linux commands and applications included in the latest CM4000 build can be found at http://www.opengear.com/faq233.html

More details on the Linux commands can found online at:

http://en.tldp.org/HOWTO/HOWTO-INDEX/howtos.html

http://www.faqs.org/docs/Linux-HOWTO/Remote-Serial-Console-HOWTO.html

http://www.stokely.com/unix.serial.port.resources/serial.switch.html

The CM4000 also embodies the *okvm* console management software. This is GPL code and the full source is available from http://okvm.sourceforge.net.

The CM4000 BIOS (boot loader code) is a port of uboot which is also a GPL package with source openly available.

The CM4000 CGIs (the html code, xml code and web config tools for the Console Manager) are proprietary to Opengear, however the code will be provided to customers, under NDA.

Also inbuilt in the CM4000 is a Port Manager application and Configuration tools as described in Chapters 11 and 12. These both are proprietary to Opengear, but open to customers (as above).

Appendix B Hardware Specifications

FEATURE	VALUE
Dimensions	CM4148: 17 x 8.5 x 1.75 in (43.2 x 21. x 4.5 cm) CM4116: 17 x 8.5 x 1.75 in (43.2 x 21. x 4.5 cm) CM4008: 8.2 x 4.9 x 1.2 in (20.8 x 12.6 x 4.5 cm)
Weight	CM4148: 2.7 kg (6 lbs) CM4116: 2.7 kg (6 lbs) CM4008: 1.0 kg (2.2 lbs)
Ambient operating temperature	5°C to 50°C (41°F to 122°F)
Non operating storage temperature	-30°C to +60°C (-20°F to +140°F)
Humidity	5% to 90%
Power	External 100-240V AC, 50/60 Hz
Power Consumption	Less than 20W
CPU	Micrel KS8695P controller
Memory	CM4148: 64MB SDRAM 16MB Flash CM4116: 64MB SDRAM 16MB Flash CM4008: 16MB SDRAM 8MB Flash
Serial Connectors	CM4148: 48 RJ-45 RS-232 serial ports CM4116: 16 RJ-45 RS-232 serial ports CM4008: 8 RJ-45 RS-232 serial ports All: 1 DB-9 RS-232 console/ modem serial port

Serial Baud Rates	RJ45 ports - 2400 to 230,400bps) DB9 port - 2400 to 115,200 bps
Ethernet Connectors	1 RJ-45 10/100Base-T Ethernet port

Appendix C Safety & Certifications

Please take care to follow the safety precautions below when installing and operating the CM4000:

- Do not remove the metal covers. There are no operator serviceable components inside. Opening or removing the cover may expose you to dangerous voltage which may cause fire or electric shock. Refer all service to Opengear qualified personnel
- To avoid electric shock the power cord protective grounding conductor must be connected through to ground.
- Always pull on the plug, not the cable, when disconnecting the power cord from the socket.

Do not connect or disconnect the CM4000 during an electrical storm. Also it is recommended you use a surge suppressor or UPS to protet the equipmet from transients.

FCC Warning Statement

This device complies with Part 15 of the FCC rules. Operation of this device is subject to the following conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference that may cause undesired operation.

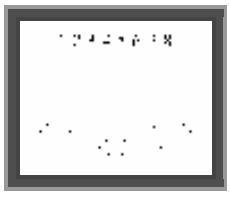
Appendix D Connectivity and Serial I/O

Pinout standards exist for both DB9 and DB25 connectors; however there are not pinout standards for serial connectivity using RJ45 connectors. Most console servers and serially managed servers/ router/ switches/ PSUs have adopted their own unique pinout; so custom connectors and cables may be required to interconnect your CM4000.

In an endeavor to create some move to standardization, Opengear products all use the same RJ45 pinout as used by Avocent and Equinox.

Serial Port Pinout

The 8/16/48 RJ45 connectors on the CM4008/4116/4148 unit have the following pinout:

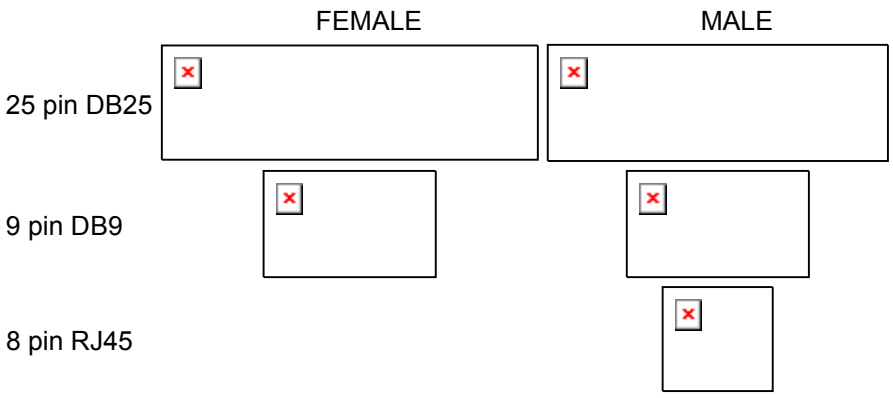


Pin	Signal	Direction	RS232 Signal Description
1	RTS	Output	Request To Send
2	DSR	Input	Data Set Ready
3	DCD	Input	Data Carrier Detect
4	RXD	Input	Receive Data
5	TXD	Output	Transmit Data
6	GND	NA	Ground
7	DTR	Output	Data Terminal Ready
8	CTS	Input	Clear To Send

The LOCAL (console/modem) port on the CM4000 uses a standard DB9 connector. The RS232 pinout standards for the DB9 (and DB25) connectors are tabled below:

DB25	SIGNAL	DB9	DEFINITION
1			Protective Ground
2	TXD	3	Transmitted Data
3	RXD	2	Received Data
4	RTS	7	Request To Send
5	CTS	8	Clear To Send
6	DSR	6	Data Set Ready
7	GND	5	Signal Ground
8	CD	1	Received Line Signal Detector

9			Reserved for data set testing
10			Reserved for data set testing
11			Unassigned
12	SCF		Secndry Rcvd Line Signl Detctr
13	SCB		Secondary Clear to Send
14	SBA		Secondary Transmitted Data
15	DB		Transmisn Signal Timng
16	SBB		Secondary Received Data
17	DD		Receiver Signal Element Timing
18			Unassigned
19	SCA		Secondary Request to Send
20	DTR	4	Data Terminal Ready
21	CG		Signal Quality Detector
22		9	Ring Indicator
23	CH/CI		Data Signal Rate Selector
24	DA		Transmit Signal Element Timing
25			Unassigned



Connectors included in CM4000

Each CM4000 ships with a “cross-over” and a “straight” RJ45-DB9 connector for connecting to other vendor’s products:

Part # 319000

DB9F-RJ45S straight connector

WIRING TABLE

	DB9F		RJ45	
RTS	7	——	1	RTS
DSR	6	——	2	DSR
DCD	1	——	3	DCD
RXD	2	——	4	RXD
TXD	3	——	5	TXD
GND	5	——	6	GND
DTR	4	——	7	DTR
CTS	8	——	8	CTS
RI	9			

Part # 319001

DB9F-RJ45S cross-over connector

WIRING TABLE

	DB9F		RJ45	
CTS	8	——	1	RTS
DTR	4	——	2	DSR
DTR	4	——	3	DCD
TXD	3	——	4	RXD
RXD	2	——	5	TXD
GND	5	——	6	GND
DSR	6	——	7	DTR
DCD	1	——	7	DTR
RTS	7	——	8	CTS
RI	9			

Other available connectors and adapters

Opengear also supplies a range of cables and adapters that will enable you to easily connect to the more popular servers and network appliances. More detailed information can be found online at http://www.opengear.com/cabling.html

- Local/Console connection

These adapters connect the CM4000 LOCAL/Console port (via standard UTP Cat 5 cable) to modem devices (for out of band access):

319000	DB9F to RJ45 straight	CM4000 LOCAL Console Port to Modem
319002	DB25M to RJ45 straight	CM4000 LOCAL Console Port to Modem

- CM4000 Serial Port connection

The connectors and adapters in the table below all work with standard UTP Cat 5 cables:

319001	DB9F to RJ45 crossover	DCE Adapter - CM4000 Ports to X86 and other
319002	DB25M to RJ45 straight	DTE Adapter - CM4000 Ports

319003	DB25M to RJ45 crossover	DCE Adapter - CM4000 Ports to Sun and other
319004	DB9M to RJ45 straight	DTE Adapter - CM4000 to Netscreen and Dell
319005	DB25F to RJ45 crossover	DCE Adapter - CM4000 to Cisco 7200 AUX
440016	5ft Cat5 RJ-45 to RJ-45 cables	Extension cables
449016	RJ-45 Plug to RJ-45 Jack	Adapter for Cisco console

Appendix E Hardware Test

This section describes the Loopback Test facilities built into the CM4000 code. When undertaking a Loopback Test, each of the serial ports loop data transmitted to data received, RTS to CTS, and DTR to DSR + DCD. The loopback program senses that data sent is received properly and that signals set and received properly. The Loopback Test also undertakes an Ethernet loopback that senses the data transmitted is received properly.

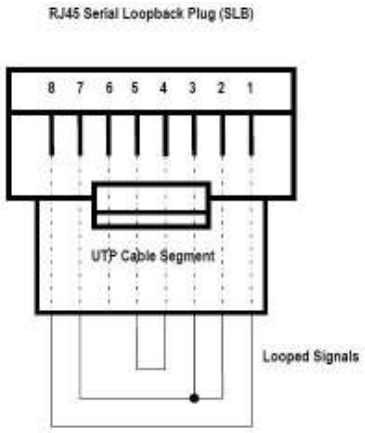
To undertake these tests you must have at hand:

- CM4000 unit (CM4008,CM4116 or CM4148)
- Terminal device (*e.g.* Windows PC and HyperTerminal program)
- Serial console cabling *e.g.* UTP Cat5 cable (#440016), DB-9 to RJ45 DTE adapter (#319000) and DB-9 to RJ45 DCE adapter (#319001)
- Custom made R-45 serial loopback plugs (SLB)
- Custom made RJ-45 Ethernet loopback plug (ELB)

SLB Serial Loopback

Signal wiring on custom made SLB loopback plug:

- Wire RTS to CTS (1 to 8)
- Wire DSR to DCD to DTR (2 to 3 to 7)
- Wire RXD to TXD (4 to 5)

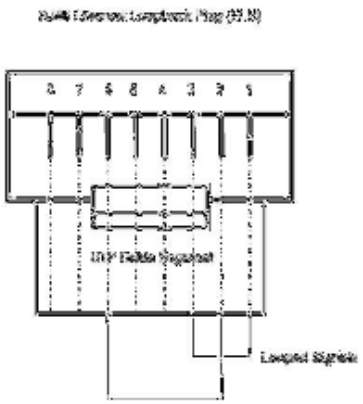


The RJ-45 Serial Modular Jack pinout is:

1	RTS
2	DSR
3	DCD
4	RXD
5	TXD
6	GND
7	DTR
8	CTS

ELB Ethernet Loopback

Signal Wiring on Custom made loopback plug:



- Wire TXD+ to RXD+ (1 to 3)
- Wire TXD- to RXD- (2 to 6)

The RJ-45 Ethernet modular jack pinout is:

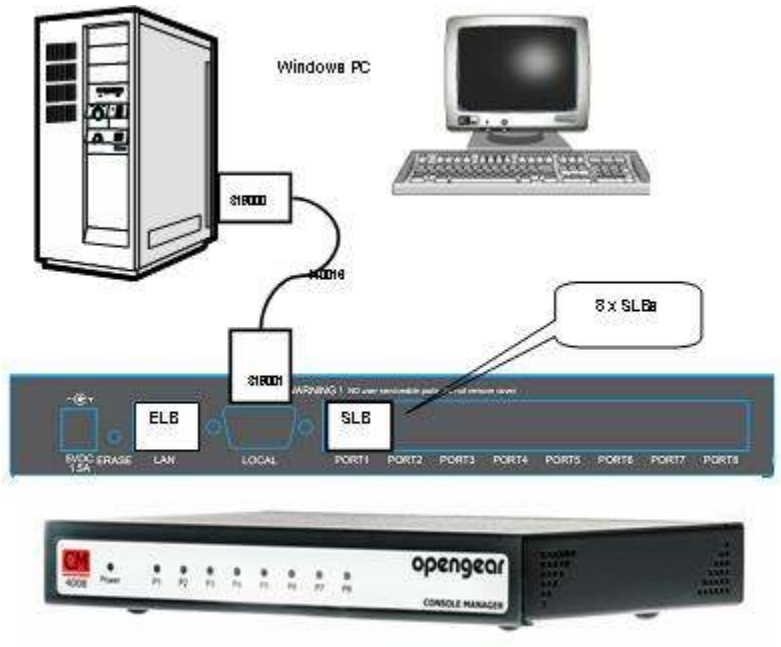
1	TXD+
2	TXD-
3	RXD+
4	NC
5	NC
6	RXD-
7	NC
8	NC

Test Procedure

- Power up the CM4000 and you should observe the LEDs P1 through P8 light up in sequence
- Configure the serial connection of the “terminal” device/program you are using to 9600bps, 8 data bits, no parity and one stop bit
- Plug a serial cable between the CM4000 local DB-9 port and terminal device. If you are using “HyperTerminal” or a similar program running on a Windows PC as the terminal device, then the cable is made up from a Cat5 UTP (440016) cable and two DB-9 to RJ-45 adapters (319000 and 319001)
- Log on to the CM4000 by pressing ‘return’ a few times. The CM4000 will request a user name and password. The user name is ‘root’ and the password is ‘default’. You should now see the command line prompt which is a hash (#)

For CM4008:

- Install the ELB on the Ethernet RJ45 socket and an SLB plug onto each serial RJ-45 sockets



- To invoke the inbuilt loopback diagnostics:

  Type in *loopback –e eth0 /dev/port0[1-8]*
  Then press '*return*'

The screen will show 8 columns for serial loopback and one for Ethernet.

1	2	3	4	5	6	7	8	E	
-	-	-	-	-	-	-	-	-	*(- is not looped)*
L	L	L	L	L	L	L	L	L	*(L is looped)*
S	S	S	S	S	S	S	S	S	*(S is too little data received)*
C	C	C	C	C	C	C	C	C	*(C is corrupt data received)*
D	D	D	D	D	D	D	D	D	*(DTR set but not sensed)*
R	R	R	R	R	R	R	R	R	*(RTS set but not sensed)*

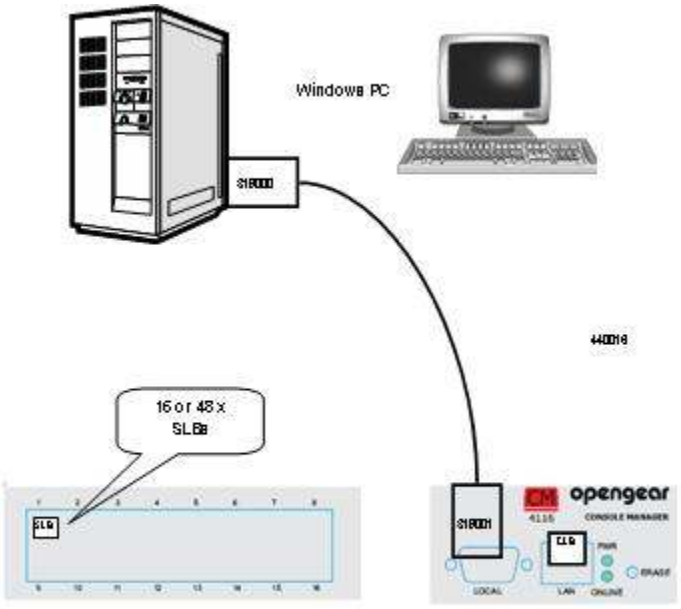
This will test port 1 through 8 and will repeat indefinitely.

The test can be terminated by pressing Ctrl C.

A successful test must have 'L' active in each column.

For CM4116/ CM4148:

- Install the ELB on the Ethernet RJ45 socket and an SLB plug onto each serial RJ-45 sockets



- To invoke the inbuilt loopback diagnostics:

  Type in *loopback –e eth0 /dev/port0[1-9]*
  Then press '*return*'

The screen will show 9 columns for serial loopback and one for Ethernet.

1	2	3	4	5	6	7	8	9	E	
-	-	-	-	-	-	-	-	-	-	*(- is not looped)*
L	L	L	L	L	L	L	L	L	L	*(L is looped)*
S	S	S	S	S	S	S	S	S	S	*(S is too little data received)*
C	C	C	C	C	C	C	C	C	C	*(C is corrupt data received)*
D	D	D	D	D	D	D	D	D	D	*(DTR set but not sensed)*
R	R	R	R	R	R	R	R	R	R	*(RTS set but not sensed)*

This will test port 1 through 9.To test ports 10 through 16 on the CM4116 you need to type -

loopback –e eth0 /dev/port1[0-6]

The screen will then show 7 columns for ports 10 through 16 and one for Ethernet.

As the CM4148 has 48 ports you need to test ports, 1-9, 10-19, 20-29, 30-39, 40-48 in separate blocks.

For ports 10 through 19, type in –

loopback –e eth0 /dev/port1[0-9]

For ports 20 through 29, type in –

loopback –e eth0 /dev/port2[0-9]

For ports 30 through 39, type in –

loopback –e eth0 /dev/port3[0-9]

For ports 40 through 48, type in –

loopback –e eth0 /dev/port4[0-8]

The test will repeat indefinitely.

The test can be terminated by pressing Ctrl C.

A successful test must have ‘L’ active in each column.

Appendix F Terminology

TERM	MEANING
Authentication	Authentication is the technique by which a process verifies that its communication partner is who it is supposed to be and not an imposter. Authentication confirms that data is sent to the intended recipient and assures the recipient that the data originated from the expected sender and has not been altered on route.
BOOTP	Bootstrap Protocol. A protocol that allows a network user to automatically receive an IP address and have an operating system boot without user interaction. BOOTP is the basis for the more advanced DHCP
Certificates	A digitally signed statement that contains information about an entity and the entity's public key, thus binding these two pieces of information together. A certificate is issued by a trusted organization (or entity) called a Certification Authority (CA) after the CA has verified that the entity is who it says it is.
Certificate Authority	A Certificate Authority is a trusted third party, which certifies public key's to truly belong to their claimed owners. It is a key part of any Public Key Infrastructure, since it allows users to trust that a given public key is the one they wish to use, either to send a private message to its owner or to verify the signature on a message sent by that owner.
Certificate Revocation List	A list of certificates that have been revoked by the CA before they expired. This may be necessary if the private key certificate has been compromised or if the holder of the certificate is to be denied the ability to establish a connection to the CM4000.
CHAP	Challenge-Handshake Authentication Protocol (CHAP) is used to verify a user's name and password for PPP Internet connections. It is more secure than PAP, the other main authentication protocol.
DHCP	Dynamic Host Configuration Protocol. A communications protocol that assigns IP addresses to computers when they are connected to the network.
DNS	Domain Name System that allocates Internet domain names and translates them into IP addresses. A domain name is a meaningful and easy to remember name for an IP address.

DUN	Dial Up Networking
Encryption	The technique for converting a readable message (plaintext) into apparently random material (ciphertext) which cannot be read if intercepted. The proper decryption key is required to read the message.
Ethernet	A physical layer protocol based upon IEEE standards
Firewall	A network gateway device that protects a private network from users on other networks. A firewall is usually installed to allow users on an intranet access to the public Internet without allowing public Internet users access to the intranet.
Gateway	A machine that provides a route (or pathway) to the outside world.
Hub	A network device that allows more than one computer to be connected as a LAN, usually using UTP cabling.
Internet	A worldwide system of computer networks - a public, cooperative, and self-sustaining network of networks accessible to hundreds of millions of people worldwide. The Internet is technically distinguished because it uses the TCP/IP set of protocols.
Intranet	A private TCP/IP network within an enterprise.
Key lifetimes	The length of time before keys are renegotiated
LAN	Local Area Network
LDAP	The Lightweight Directory Access Protocol (LDAP) is based on the X.500 standard, but significantly simpler and more readily adapted to meet custom needs. The core LDAP specifications are all defined in RFCs. LDAP is a protocol used to access information stored in an LDAP server.
LED	Light-Emitting Diode
MAC address	Every piece of Ethernet hardware has a unique number assigned to it called it's MAC address. Ethernet is used locally to connect the CM4000 to the Internet, and it may share the local network with many other appliances. The MAC address is used by the local Internet router in order to direct CM400 traffic to it rather than somebody else in the local area. It is a 48-bit number usually written as a series of 6 hexadecimal octets, *e.g.* 00:d0:cf:00:5b:da. A CM4000 has a MAC address listed on a label underneath the device.

MSCHAP	Microsoft Challenge Handshake Authentication Protocol (MSCHAP) is authentication for PPP connections between a computer using a Microsoft Windows operating system and a network access server. It is more secure than PAP or CHAP, and is the only option that also supports data encryption.
NAT	Network Address Translation. The translation of an IP address used on one network to an IP address on another network. Masquerading is one particular form of NAT.
Net mask	The way that computers know which part of a TCP/IP address refers to the network, and which part refers to the host range.
NFS	Network File System is a protocol that allows file sharing across a network. Users can view, store, and update files on a remote computer.
NTP	Network Time Protocol (NTP) used to synchronize clock times in a network of computers
PAP	Password Authentication Protocol (PAP) is the usual method of user authentication used on the internet: sending a username and password to a server where they are compared with a table of authorized users. Whilst most common, PAP is the least secure of the authentication options.
PPP	Point-to-Point Protocol. A networking protocol for establishing simple links between two peers.
RADIUS	The Remote Authentication Dial-In User Service (RADIUS) protocol was developed by Livingston Enterprises as an access server authentication and accounting protocol. The RADIUS server can support a variety of methods to authenticate a user. When it is provided with the username and original password given by the user, it can support PPP, PAP or CHAP, UNIX login, and other authentication mechanisms.
Router	A network device that moves packets of data. A router differs from hubs and switches because it is "intelligent" and can route packets to their final destination.
SMTP	Simple Mail Transfer Protocol. CM4000 includes, SMTPclient, a minimal SMTP client that takes an email message body and passes it on to a SMTP server (default is the MTA on the local host).
SSH	Secure Shell is secure transport protocol based on public-key cryptography.
SSL	Secure Sockets Layer is a protocol that provides authentication and encryption services between a web server and a web browser.
TACACS+	The Terminal Access Controller Access Control System (TACACS+) security protocol is a more recent protocol developed by Cisco. It provides

	detailed accounting information and flexible administrative control over the authentication and authorization processes. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide authentication, authorization, and accounting services independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon. There is a draft RFC detailing this protocol.
TCP/IP	Transmission Control Protocol/Internet Protocol. The basic protocol for Internet communication.
TCP/IP address	Fundamental Internet addressing method that uses the form nnn.nnn.nnn.nnn.
Telnet	Telnet is a terminal protocol that provides an easy-to-use method of creating terminal connections to a network.
UTC	Coordinated Universal Time.
UTP	Unshielded Twisted Pair cabling. A type of Ethernet cable that can operate up to 100Mb/s. Also known as Category 5 or CAT 5.
VNC	Virtual Network Computing (VNC) is a desktop protocol to remotely control another computer. It transmits the keyboard presses and mouse clicks from one computer to another relaying the screen updates back in the other direction, over a network.
WAN	Wide Area Network
WINS	Windows Internet Naming Service that manages the association of workstation names and locations with IP addresses

For further technology definitions refer:
http://linux-documentation.com/en/documentation/linux-dictionary/index.html

Appendix G End User License Agreement

READ BEFORE USING THE ACCOMPANYING SOFTWARE

YOU SHOULD CAREFULLY READ THE FOLLOWING TERMS AND CONDITIONS BEFORE USING THE ACCOMPANYING SOFTWARE, THE USE OF WHICH IS LICENSED FOR USE ONLY AS SET FORTH BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT, DO NOT USE THE SOFTWARE. IF YOU USE ANY PART OF THE SOFTWARE, SUCH USE WILL INDICATE THAT YOU ACCEPT THESE TERMS.

You have acquired a product that includes Opengear ("Opengear") proprietary software and/or proprietary software licensed to Opengear. This Opengear End User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and Opengear for the installed software product of Opengear origin, as well as associated media, printed materials, and "online" or electronic documentation ("Software"). By installing, copying, downloading, accessing, or otherwise using the Software, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this EULA, Opengear is not willing to license the Software to you. In such event, do not use or install the Software. If you have purchased the Software, promptly return the Software and all accompanying materials with proof of purchase for a refund.

Products with separate end user license agreements that may be provided along with the Software are licensed to you under the terms of those separate end user license agreements.

LICENSE GRANT. Subject to the terms and conditions of this EULA, Opengear grants you a nonexclusive right and license to install and use the Software on a single CPU, provided that, (1) you may not rent, lease, sell, sublicense or lend the Software; (2) you may not reverse engineer, decompile, disassemble or modify the Software, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation; and (3) you may not transfer rights under this EULA unless such transfer is part of a permanent sale or transfer of the Product, you transfer at the same time all copies of the Software to the same party or destroy such materials not transferred, and the recipient agrees to this EULA.

No license is granted in any of the Software's proprietary source code. This license does not grant you any rights to patents, copyright, trade secrets, trademarks or any other rights with respect to the Software.

You may make a reasonable number of copies of the electronic documentation accompanying the Software for each Software license you acquire, provided that, you must reproduce and include all copyright notices and any other proprietary rights notices appearing on the electronic documentation. Opengear reserves all rights not expressly granted herein.

INTELLECTUAL PROPERTY RIGHTS. The Software is protected by copyright laws, international copyright treaties, and other intellectual property laws and treaties. Opengear and

its suppliers retain all ownership of, and intellectual property rights in (including copyright), the Software components and all copies thereof, provided however, that certain components of the Software are components licensed under the GNU General Public License (version 2), which Opengear supports. You may obtain a copy of the GNU General Public License at http:/www.fsf.org/copyleft/gpl.html. Opengear will provide source code for any of the components of the Software licensed under the GNU General Public License upon request.

EXPORT RESTRICTIONS. You agree that you will not export or re-export the Software, any part thereof, or any process or service that is the direct product of the Software in violation of any applicable laws or regulations of the United States or the country in which you obtained them.

U.S. GOVERNMENT RESTRICTED RIGHTS. The Software and related documentation are provided with Restricted Rights. Use, duplication, or disclosure by the Government is subject to restrictions set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c) (1) and (2) of the Commercial Computer Software – Restricted Rights at 48 C.F.R. 52.227-19, as applicable, or any successor regulations.

TERM AND TERMINATION. This EULA is effective until terminated. The EULA terminates immediately if you fail to comply with any term or condition. In such an event, you must destroy all copies of the Software. You may also terminate this EULA at any time by destroying the Software.

GOVERNING LAW AND ATTORNEY'S FEES. This EULA is governed by the laws of the State of Utah, USA, excluding its conflict of law rules. You agree that the United Nations Convention on Contracts for the International Sale of Goods is hereby excluded in its entirety and does not apply to this EULA. If you acquired this Software in a country outside of the United States, that country's laws may apply. In any action or suit to enforce any right or remedy under this EULA or to interpret any provision of this EULA, the prevailing party will be entitled to recover its costs, including reasonable attorneys' fees.

ENTIRE AGREEMENT. This EULA constitutes the entire agreement between you and Opengear with respect to the Software, and supersedes all other agreements or representations, whether written or oral. The terms of this EULA can only be modified by express written consent of both parties. If any part of this EULA is held to be unenforceable as written, it will be enforced to the maximum extent allowed by applicable law, and will not affect the enforceability of any other part.

Should you have any questions concerning this EULA, or if you desire to contact Opengear for any reason, please contact the Opengear representative serving your company.

THE FOLLOWING DISCLAIMER OF WARRANTY AND LIMITATION OF LIABILITY IS INCORPORATED INTO THIS EULA BY REFERENCE. THE SOFTWARE IS NOT FAULT TOLERANT. YOU HAVE INDEPENDENTLY DETERMINED HOW TO USE THE SOFTWARE IN THE DEVICE, AND OPENGEAR HAS RELIED UPON YOU TO CONDUCT SUFFICIENT TESTING TO DETERMINE THAT THE SOFTWARE IS SUITABLE FOR SUCH USE.

LIMITED WARRANTY Opengear warrants the media containing the Software for a period of ninety (90) days from the date of original purchase from Opengear or its authorized retailer.

Proof of date of purchase will be required. Any updates to the Software provided by Opengear (which may be provided by Opengear at its sole discretion) shall be governed by the terms of this EULA. In the event the product fails to perform as warranted, Opengear's sole obligation shall be, at Opengear's discretion, to refund the purchase price paid by you for the Software on the defective media, or to replace the Software on new media. Opengear makes no warranty or representation that its Software will meet your requirements, will work in combination with any hardware or application software products provided by third parties, that the operation of the software products will be uninterrupted or error free, or that all defects in the Software will be corrected.

OPENGEAR DISCLAIMS ANY AND ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OTHER THAN AS STATED HEREIN, THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT IS WITH YOU. ALSO, THERE IS NO WARRANTY AGAINST INTERFERENCE WITH YOUR ENJOYMENT OF THE SOFTWARE OR AGAINST INFRINGEMENT. IF YOU HAVE RECEIVED ANY WARRANTIES REGARDING THE DEVICE OR THE SOFTWARE, THOSE WARRANTIES DO NOT ORIGINATE FROM, AND ARE NOT BINDING ON, OPENGEAR.

NO LIABILITY FOR CERTAIN DAMAGES. EXCEPT AS PROHIBITED BY LAW, OPENGEAR SHALL HAVE NO LIABILITY FOR COSTS, LOSS, DAMAGES OR LOST OPPORTUNITY OF ANY TYPE WHATSOEVER, INCLUDING BUT NOT LIMITED TO, LOST OR ANTICIPATED PROFITS, LOSS OF USE, LOSS OF DATA, OR ANY INCIDENTAL, EXEMPLARY SPECIAL OR CONSEQUENTIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY OR OTHERWISE ARISING FROM OR IN CONNECTION WITH THIS EULA OR THE USE OR PERFORMANCE OF THE SOFTWARE. IN NO EVENT SHALL OPENGEAR BE LIABLE FOR ANY AMOUNT IN EXCESS OF THE LICENSE FEE PAID TO OPENGEAR UNDER THIS EULA. SOME STATES AND COUNTRIES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS LIMITATION MAY NOT APPLY TO YOU.

Appendix H Service and Warranty

STANDARD WARRANTY

Opengear, Inc., its parent, affiliates and subsidiaries, (collectively, "Opengear")warrant your Opengear product to be in good working order and to be free from defects in workmanship and material (except in those cases where the materials are supplied by the Purchaser) under normal and proper use and service for the period of one (1) year from the date of original purchase from an Authorized Opengear reseller. In the event that this product fails to meet this warranty within the applicable warranty period, and provided that Opengear confirms the specified defects, Purchaser's sole remedy is to have Opengear, in Opengear's sole discretion, repair or replace such product at the place of manufacture, at no additional charge other than the cost of freight of the defective product to and from the Purchaser. Repair parts and replacement products will be provided on an exchange basis and will be either new or reconditioned. Opengear will retain, as its property, all replaced parts and products. Notwithstanding the foregoing, this hardware warranty does not include service to replace or repair damage to the product resulting from accident, disaster, abuse, misuse, electrical stress, negligence, any non- Opengear modification of the product except as provided or explicitly recommended by Opengear, or other cause not arising out of defects in material or workmanship. This hardware warranty also does not include service to replace or repair damage to the product if the serial number or seal or any part thereof has been altered, defaced or removed. If Opengear does not find the product to be defective, the Purchaser will be invoiced for said inspection and testing at Opengear's then current rates, regardless of whether the product is under warranty.

RMA RETURN PROCEDURE

If this product requires service during the applicable warranty period, a Return Materials Authorization (RMA) number must first be obtained from Opengear. Product that is returned to Opengear for service or repair without an RMA number will be returned to the sender unexamined. Product should be returned, freight prepaid, in its original or equivalent packaging, to:

Opengear Service Center
7984 South Welby Park Drive #101
Salt Lake City, Utah 84084

Proof of purchase date must accompany the returned product and the Purchaser shall agree to insure the product or assume the risk of loss of damage in transit. Contact Opengear by emailing support@opengear.com for further information.

TECHNICAL SUPPORT

Purchaser is entitled to thirty (30) days free telephone support (USA ONLY) and twelve (12) months free e-mail support (world wide) from date of purchase provided that the Purchaser first register their product(s) with Opengear by filling in the on-line form http://www.opengear.com/registration.html. Telephone and e-mail support is available from 9:00 AM to 5:00 PM, Mountain Time.

Opengear's standard warranty includes free access to Opengear's Knowledge Base as well as any application notes, white papers and other on-line resources that may become available from time to time.

Opengear reserves the right to discontinue all support for products that are no longer covered by warranty.

LIMITATION OF LIABILITY

No action, regardless of form, arising from this warranty may be brought by either party more than two (2) years after the cause of action has occurred. Purchaser expressly agrees that Opengear's liability, if any, shall be limited solely to the replacement or repair of the product in accordance with the warranties specifically and expressly set forth herein. The remedies of the Purchaser are the exclusive and sole remedies available, and, in the event of a breach or repudiation of any provision of this agreement by Opengear, the Purchaser shall not be entitled to receive any incidental damages as that term is defined in Section 2-715 of the Uniform Commercial Code. Opengear waives the benefit of any rule that disclaimer of warranty shall be construed against Opengear and agrees that such disclaimers herein shall be construed liberally in favor of Opengear.

THE FOREGOING WARRANTIES ARE THE SOLE ANDEXCLUSIVE WARRANTIES GIVEN IN CONNECTION WITH THE PRODUCT AND THE HARDWARE. OPENGEAR DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, ANY WARRANTIES AS TO THE SUITABILITY OR MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. OPENGEAR DOES NOT PROMISE THAT THE PRODUCT IS ERROR-FREE OR WILL OPERATE WITHOUT INTERRUPTION. IN NO EVENT SHALL OPENGEAR BE LIABLE FOR ANY LOST OR ANTICIPATED PROFITS, OR ANY INCIDENTAL, EXEMPLARY, SPECIAL OR CONSEQUENTIAL DAMAGES, REGARDLESS OF WHETHER OPENGEAR WAS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.